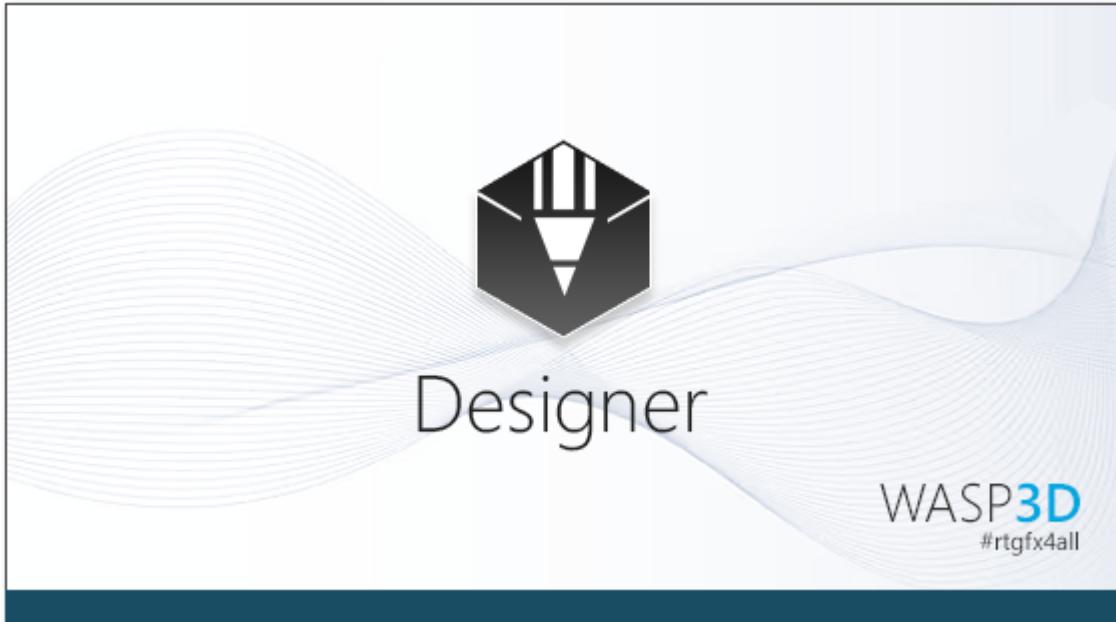


Welcome

Drone Designer, the real-time 3D TV graphics Design Software is the graphics creation environment of the WASP3D workflow. Drone Designer, the TV graphics Software unleashes a graphic artist's creative power to generate, manipulate, animate, and visualize 3D graphics with great ease. The broadcast graphics design software's highly intuitive interface minimizes the learning curve while expanding the graphical expression capabilities of design professionals.



Copyrights and Trademarks

Copyright ©2007-2024 Beehive Systems
All Rights Reserved

The software contains proprietary information of Beehive Systems; it is provided under a license agreement containing restrictions on use and disclosure and is protected by copyright law. Reverse engineering of the software is prohibited.

Due to continued product development, this information may change without notice. The information and intellectual property contained herein is confidential between Beehive Systems and the client and remains the exclusive property of Beehive Systems. If you find any problems in the documentation, please report them to us in writing. Beehive Systems does not warrant that this document is error-free.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without the prior written permission Beehive Systems.

Third-Party Trademarks

All other brand names, product names or trademarks belong to their respective holders. Microsoft DirectX, Windows ©, Windows 7© are trademarks of the Microsoft Corporation. Apple QuickTime© is a trademark of Apple Corporation.

Beehive Systems Limited.

B-37, Sector-1, Noida,
Uttar Pradesh, 201301, INDIA.
Phone: +91.120.430.8888
E-Mail: support@wasp3d.com
Website: www.wasp3d.com

End User License Agreement

This End User License Agreement ("License") is a legal agreement between you and Beehive Systems ("Beehive") regarding Beehive's software and the accompanying documentation (collectively, the "Software"). Carefully read the following license. Using this software indicates your acceptance of this license. If you do not accept or understand the terms of this license, you should not install the software and should promptly return this product to the place of purchase.

License Grant

Subject to the restrictions set forth below, this License grants you a non-exclusive, perpetual license to (a) install the Software on only one computer; (b) use or authorize use of the Software on only one computer; (c) make one copy of the Software, in machine-readable form, solely for backup purposes; provided you include all copyright and other proprietary rights notices on the copy and (d) transfer the

Software and this License to another party if the other party agrees to accept the terms and conditions of this License. If you transfer the Software, you must at the same time either transfer all copies to the same party or destroy any copies not transferred. If you transfer possession of any copy of the Software to another party, your License is automatically terminated. Some features of the Software may be locked and require that you activate, either for free or subject to payment of additional license fees, the features before you may use them. Additionally, Beehive may, either for free or subject to payment of additional license fees, grant you a license for plug-ins to the Software produced or distributed by Beehive (the "Plug-ins"). Your use of any such features and any of the Plug-ins shall be subject to the terms of this Agreement, unless accompanied by another license agreement, in which case the Plug-ins license agreement shall prevail.

License Restrictions

You may not, or allow any third party to, (a) rent, lease, sell, loan or otherwise transfer the Software or any of your rights and obligations under this License; (b) install the Software on a network for use by multiple users, unless each user has purchased a license; (c) reverse-engineer, de-compile or disassemble the Software or hardware in whole or in part; (d) remove or destroy any copyright notices or other proprietary markings of the Software or any third-party software; (e) modify or adapt the Software, merge the Software into another program or create derivative works based on the Software; (f) make copies of or distribute, for profit or otherwise, the Software, except as expressly provided above; (g) make any alteration, modification, connection, disconnection, improvement or adjustment of any kind to or use the Software except as explicitly contemplated in the enclosed documentation and in this License and (h) sub-license, transfer or assign this License or any of the rights and obligations granted under this License, except as explicitly contemplated in this License. Any purported transfer or assignment will be void.

Ownership

The license granted here under does not constitute a transfer or sale of ownership rights in or to the Software. Except for the license rights granted above, Beehive retains all rights, title and interest in and to the Software including all intellectual property rights therein.

Third Party Proprietary Property

This Software may contain the proprietary property of others, which has been licensed to Beehive. Your use of the Software is expressly subject to your agreement not to remove any copyright notices or other proprietary markings of the third party software.

Security

You acknowledge and agree that in order to protect the integrity of certain third party content, Beehive and/or its licensors may provide for Software security related updates that will be downloadable for installation on your computer. Such security related updates may impair the Software (and any other software on your computer which specifically depends on the Software) including disabling your ability to copy and/or play "secure" content, i.e. content protected by digital rights management. In such an event, Beehive and/or its licensors shall use reasonable efforts to post notices promptly on Wasp3D web site explaining the security update and providing instructions to end-users for obtaining new versions or further updates of the Software that restore access to secure content and related features.

Updates

You acknowledge and agree that Beehive may automatically check the version of the Software and/or its components that you are using and may provide updates or fixes to the Software that may be automatically downloaded to your computer. Updates provided after the expiry of the Limited Warranty period set forth in Section 8 below are not covered by any warranty, express, implied, or statutory.

Term and Termination

This License shall be effective upon installation of the Software and shall terminate upon the earlier of (a) your failure to comply with any term of this License; (b) return, destruction or deletion of all copies of the Software in your possession or (c) transfer of the Software and this License to another party in accordance with Section 1 (d) Beehive's rights and your obligations shall survive termination of this License.

Limited Warranty

Beehive warrants to the original licensee that the Software, as delivered, will perform in accordance with the accompanying documentation for a period of 30 days from the date of the original purchase ("Limited Warranty"). Beehive's entire liability and your exclusive remedy for breach of the preceding Limited Warranty shall be, at Beehive's sole option, repair or replacement of the Software that does not meet the warranty set forth herein and that is returned to Beehive. This limited warranty shall be void if failure of the Software has resulted from any accident, abuse, misuse or misapplication by you.

Limitation of Liability

In no event will Beehive or its suppliers be liable to you for any damages, claims or costs whatsoever or any consequential, indirect, incidental damages, or any lost profits or lost savings, even if a Beehive representative has been advised of the possibility of such loss, damages, claims or costs or for any claim by any third party.

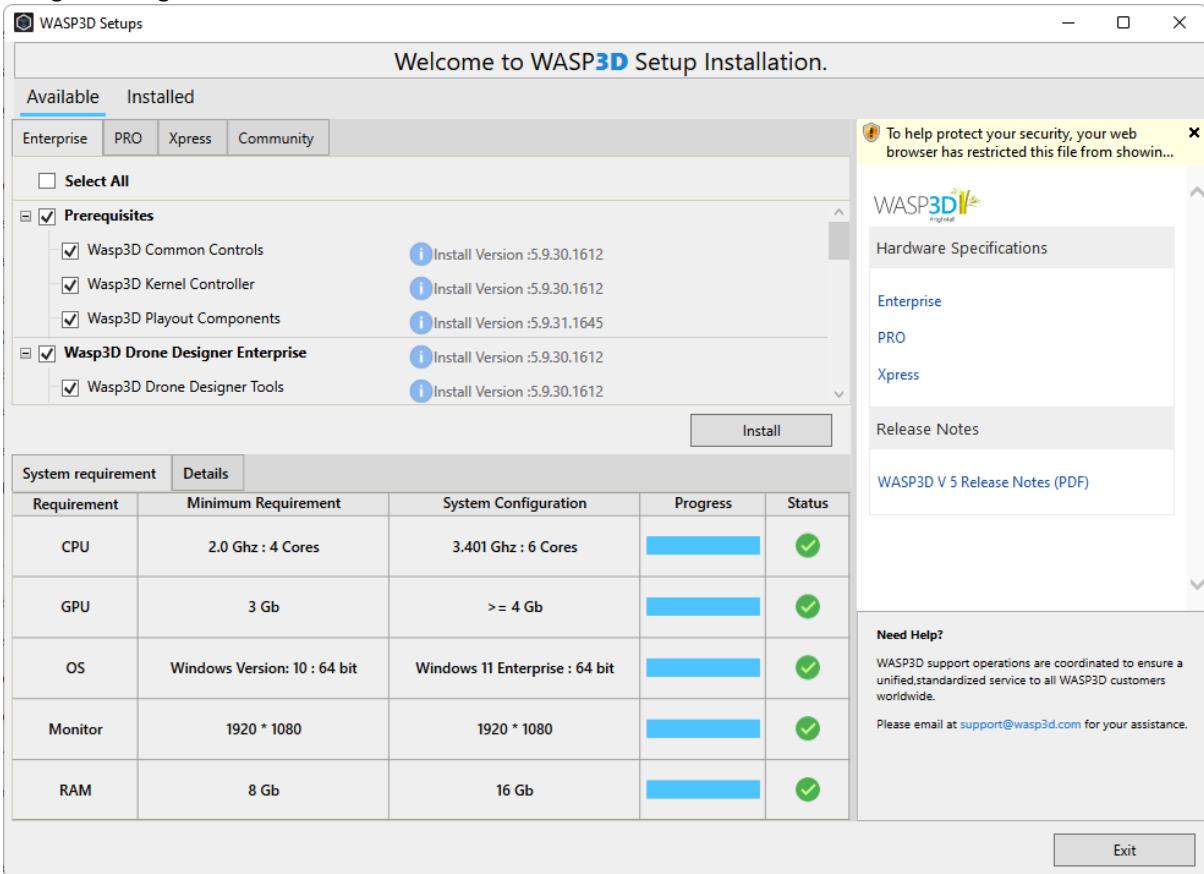
General

If any part of this Agreement is found void and unenforceable, it will not affect the validity of the balance of this Agreement, which shall remain valid and enforceable according to its terms. This Agreement shall not prejudice the statutory rights of any party dealing as a consumer. Updates may be licensed to you by Beehive with additional or different terms. This is the entire agreement between Beehive and you relating to the Software and it supersedes any prior representations, discussions, undertakings, communications or advertising relating to the Software.

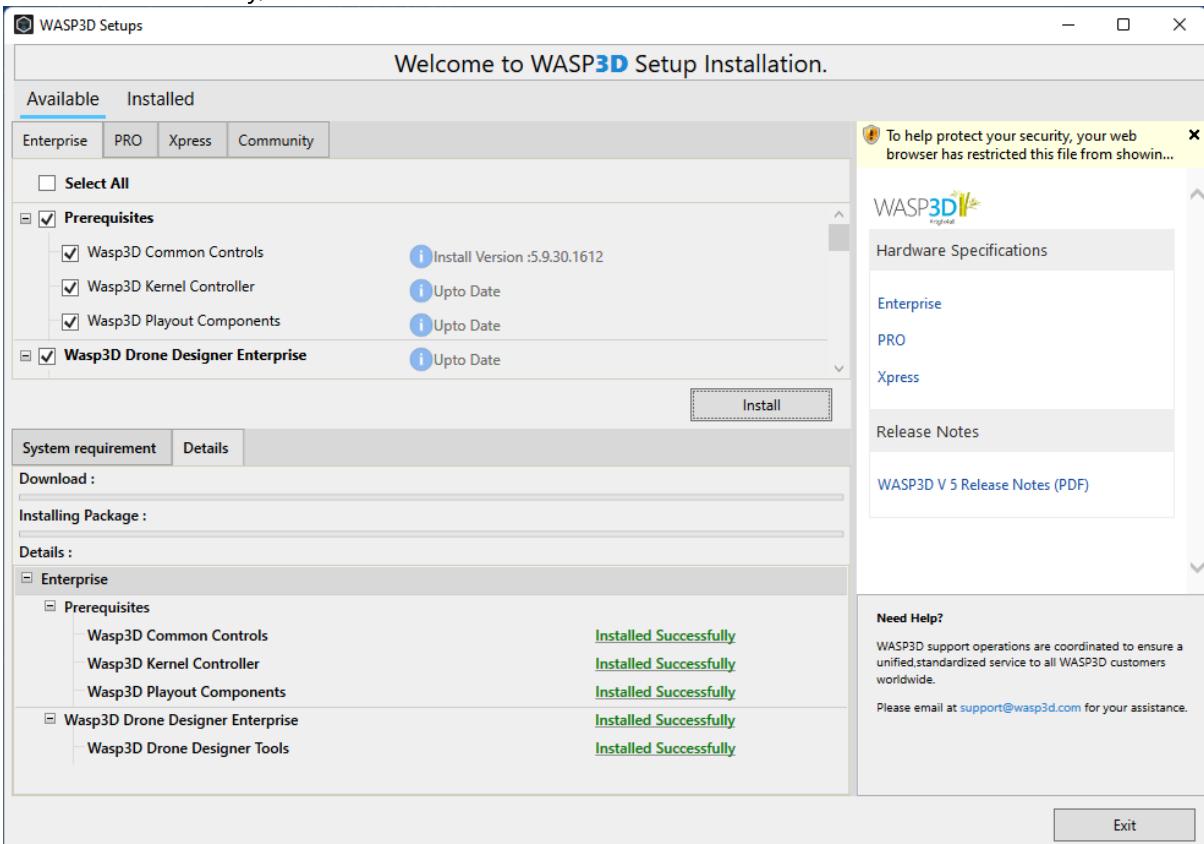
Installing Drone Designer Application Steps

1. Run the "**WASP3D Setups Installer (x64).exe**" from the extracted Setup folder.

2. This will check the Hardware Prerequisites, once the minimum requirement status is green, you are good to go with the installation further.



3. Installation will start and progress will be seen in the Details tab. Once all the modules are Installed Successfully, Exit the installer.



First Steps

1. Make sure the following applications are running in Task Bar, if not double click on the below

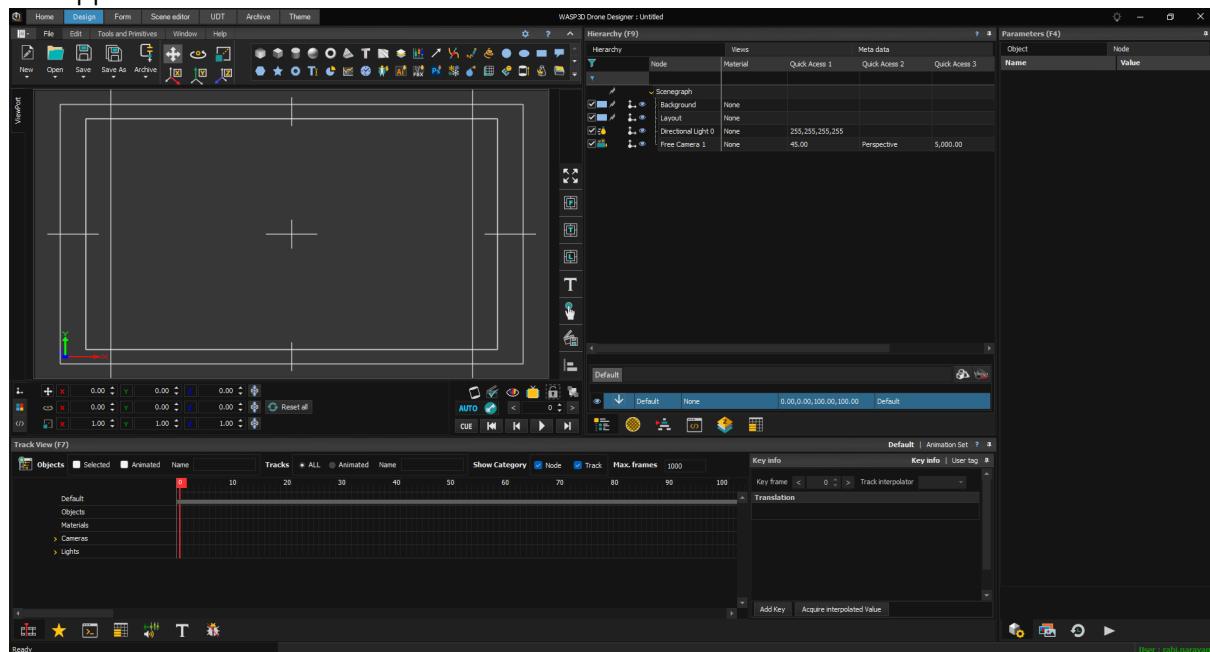


Icon. WASP3D Kernel Controller Service



2. Run the WASP3D Drone Designer (x64)

3. The application will be about to launch.

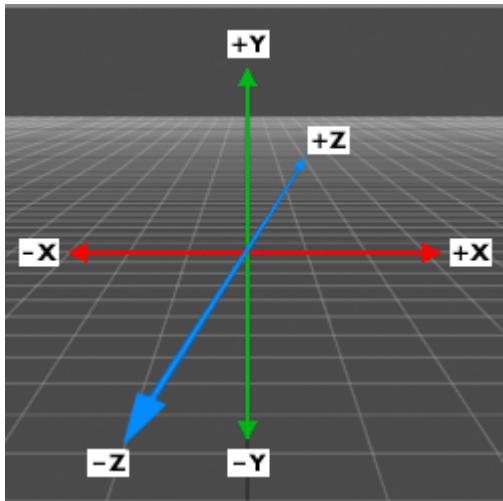


To quit the installation process, click the Cancel button, and click the Back button to get back to the previous screen.

Drone Designer Concepts

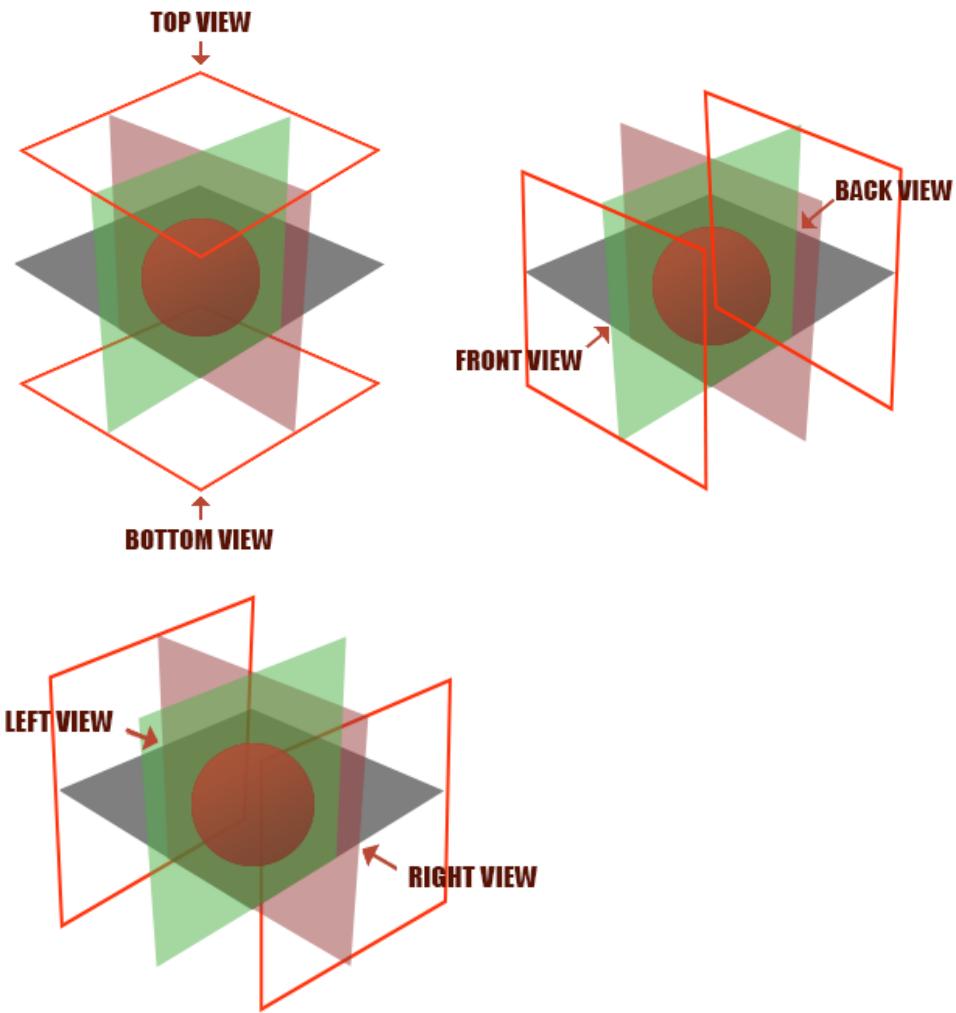
Cartesian Coordinate System

Drone Designer Cartesian Coordinate System uses a Left-handed coordinate system.



SceneGraph Viewport

SceneGraph Viewport is the region of a frame that defines the visibility extents of a scene, objects outside this region will not appear even if present. The Drone Designer presents a single Viewport view that the user can toggle to any of the orthographic view (top, front, etc.) or to a perspective view (camera).



Starting the Drone Designer

To start the Drone Designer application:

1. Click Start button or press WINDOWS from the keyboard.
2. Point to All Programs, and then click Wasp3D.
3. Then click Drone Designer.

To exit the Drone Designer application:

- On the File menu, click Exit.
- A dialog box appears asking your confirmation depending on following scenarios:
 - a. In case of any unsaved scene
 - i. Click 'Yes' to save it then exit.
 - ii. Click 'No' to exit without saving the scene.
 - iii. Click 'Cancel' to keep the application running.
 - b. In case of closing the application without making any change, it asks for your **confirmation**.
 - i. Click 'Yes' to exit. Or
 - ii. Click 'No' to keep it running.



Alternatively, double-click  (Drone Designer shortcut icon) at the Desktop to start the application. Click  **Exit** to exit the application.

Forum - Explore newly launched Drone Designer Community Edition

The Forum tab allows the users to directly connect to the Wasp3D forum using a unique ID and password or Facebook ID. Users can download the WASP3D Community Edition v3.4 and share their ideas, experiences, work etc.

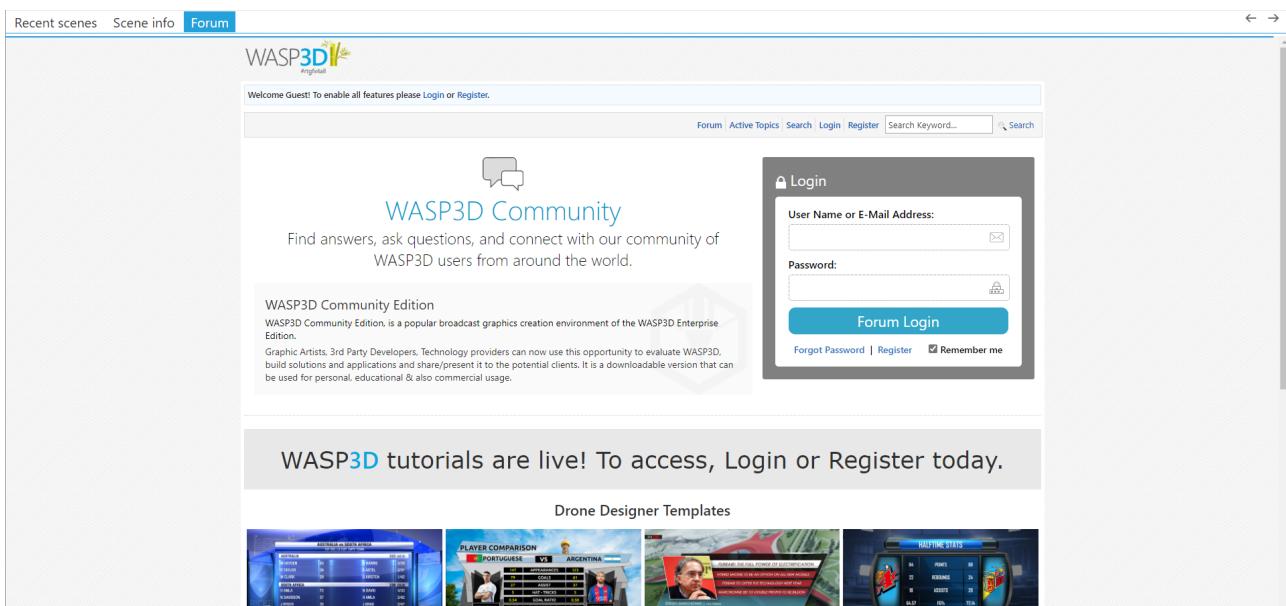
WASP3D launches Drone Designer Community Edition, which includes:

- Training Software
- Learning Resources: User Manual and Sample Templates
- Support from in-house experts...

Sign-in to our Forum now and download Drone Designer Community Edition from the Downloads section.

With Community Edition, create your own real-time 3D graphic templates & render outputs and upload links from YouTube or Vimeo as desired.

For queries, please contact us at Learnwasp3d@beesys.com



Recent Scene

When you open Drone Designer, the Home tab is first screen that appears in the application. This tab displays ten most recent templates and Scenegraph (SGs) opened in your application, information about the current Scenegraph (SG) and list of activities. The Home tab displays this information in three different sections:

Recent SG - Displays Recent Templates and Scene Graphs opened on your machine.

Recent SG screen shows the list (as thumb nail) of ten recently opened Scenegraph and WASP templates. It also provides an option to search for templates available on network drives.

Search for Templates

The Search for Template option allows you to search for a specific template available on the computers connected in your network. The search can be based on specific word(s) part of word or using wild card character “*”.

To search for a template:

Type word(s) or character(s) in the text box.

Click Search to locate files that contains word(s) or character(s) entered in the Search text box. The searched list will appear in the form of a grid just below the form.

Name of the file is displayed as link. You can click the Name link and open the template file in your machine for use.

The screenshot displays the 'Recent SG' interface. At the top, there are tabs for 'Recent scenes', 'Scene info', and 'Forum'. Below this, under 'Recent WASP template', are four thumbnail images of templates: 'News To Lower Third Swap', 'News To Lower Third Swap', 'Frame01.wspx', and 'Frame01.wspx'. A navigation bar at the bottom shows pages 1, 2, and 3. On the right side, there is a search bar labeled 'Search for template' with a magnifying glass icon. The main area contains a grid of 24 template thumbnails, each with a letter (E or P) and a file name. The templates include various designs like 'MATCH RESULTS + SCORE (SPL...)', '2 windows.wspx', '2 INDICES.wspx', 'Sports_Template1.wspx', '2 IMAGES (Image + Map) + M...', '2 IMAGES (Image + Map) + M...', 'News Tv Lower Third.wspx', '2 OPTION POLL QUESTION.wspx', 'Frame03.wspx', '4 LINE BULLETS.wspx', 'ONE LIVE VIDEO + 2 IMAGE (1 ...', 'Virtual SetScene_P.wspx', 'SINGLE TEMPERATURE DATA.w...', 'Training_Part1.wspx', '3 OPTION POLL QUESTION.wspx', 'LIVE VIDEO + 1 IMAGE (With o...', 'Frame05.wspx', 'Matte Scene_P.wspx', 'IMAGE + 4 LINE BULLETS.wspx', 'CITY WEATHER (NOW & FORE...', 'LIVE VIDEO + 1 IMAGE (With No...', 'ONE LIVE VIDEO + 2 IMAGE (G...', 'STORY STRIP (Multi-Line).wsp...', '3 LINE PROGRAM SCHEDULE.wsp...', 'COMPANY STOCK COMPARIS...', 'CURRENCY MARKET.wspx', 'SINGLE LINE Generic Design - ...', 'QUOTE Text - Multi-Line QUOT...', 'Two LINE GENERIC Tag.wspx', and 'ProgPg_2_P_Scene_P.wspx'.

Scenegraph Info - Provides information about the current scene graph

The Scenegraph Info section displays complete information about the scene you have designed like name of the user who has modified the scene, date on which scene has been modified, its video format, details of objects, gestures, texture, material A material is a design that you assign to the surface or face of an object so that it appears a certain way when rendered. Materials affect the color of objects, their glossiness, their opacity, and texture. etc. used in the scene. The Scene Information screen displays example information about a scene created:

Metadata Information

The Scene information screen provides information about the scene created:

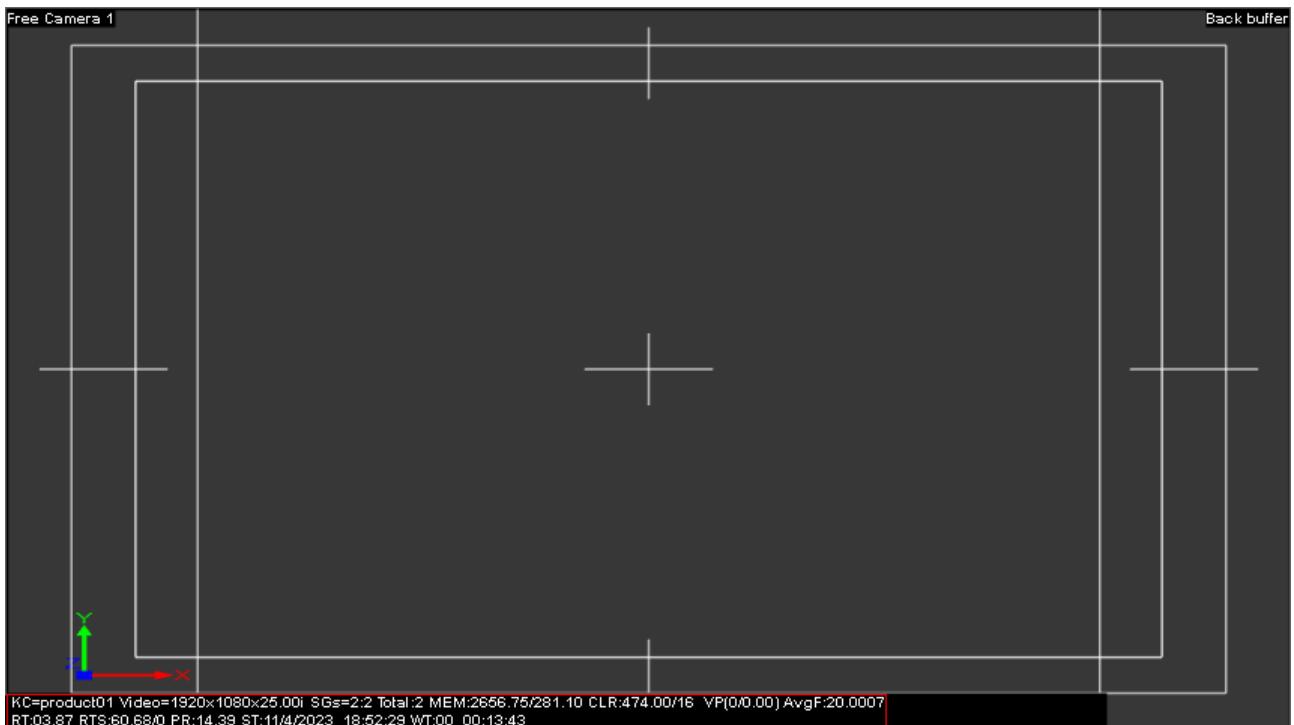
- Assets - Lists the assets like textures, shapes, models etc. used within a scene along with their paths.
- Page Controllers - Lists the controllers used within a scene.
- Action Sets - Lists the Action sets made in the scene.
- Add-in - Shows the Addins and their respective paths used in the scene.
- Variables - Lists the variables created in the scene.
- User Tags - Lists the user tags that are being used for wiring of object parameters with the real-time data.

The screenshot shows the 'Scene info' tab selected in the top navigation bar. The page is divided into several sections:

- Metadata information:** Displays basic scene details: Default channel (Any), Modified by (rabi.narayan), and Scene ID.
- Video format:** Shows video settings: Format (1080i 50Hz (1920x1080)), Frame rate (25), and Interface (true).
- Page controller:** Lists page controller names.
- Assets:** Lists asset names and types.
- Action set:** Lists action set names.
- Addin:** Shows addin instance data, including Local - Instance 1, Table Name, Default, Summary, and Variables. It also lists Local - WString with Variable Name and Value.
- User tag:** Shows user tag information.

View Port

A View-Port is a designated area on a display screen where you can observe design elements and information. Within the Drone Designer application, the view-port is divided into two distinct sections.



- Render Statistics
- Right-click drop down options

Render Statistics

To display/hide set of information, press "ALT + C" or "ALT + C + C" from the keyboard. It portrays various render statistics information at bottom of the Scene Graph, which helps users to view all render statistics information collectively in the Statistics bar.

Engine Data

- KC (Server Name): This displays the name of the web server that the software is currently connected to. It helps identify the server you are working with.
- Video (Video Characteristics): This section provides information about the video being processed:
 - Dimensions: It shows the width and height of the video in pixels.
 - Frame Rate: It specifies the number of frames displayed per second. The video can be either interlaced (i) or progressive (p), depending on the method used to display frames

Object Data

- SGs (SceneGraphs): It shows the number of loaded SGs, which are elements that make up a scene. This also includes a ratio, indicating how many SGs are currently loaded compared to the total SGs that need to be loaded. For example, if you have 10 SGs in total and 3 are currently loaded, it will display "SGs: 3:10".
- Total SGs: This number represents the total count of SGs that have been opened since you started using Drone Designer.
- MEM (Memory Usage): This section provides information about memory usage:
 - Initial: It shows the initial memory usage.
 - Delta: It indicates changes in memory usage over time.

- CLR (Common Language Runtime): This displays information related to the memory used by the Common Language Runtime (CLR) or .NET memory.
- VP (Virtual Memory Pages): It shows the count of virtual memory pages used.
- AvgF (Average Frame Rate): This value represents the average frame rate of the rendering engine, calculated based on the minimum and maximum render time ranges.

Render Stats

- MS (Miss-Sets): This indicates the number of Miss-Sets. A Miss-Set occurs when an I/O card drops a frame or when rendering a frame takes longer than the expected display time. An increasing Miss-Set count suggests that the scene rendering time is too high. This can happen when multiple scenes are loaded, and the engine struggles to render them in the required time.
- RT (Frame Render Time): This shows the time taken to render the current frame in milliseconds. The time varies depending on the video standard (e.g., 20 milliseconds for 1 field in PAL, or 40 milliseconds for 1 frame).

The formula for RT is:

- $RT = 1000 \text{ (1000 milliseconds in 1 second)} / (\text{Frame Rate} * \text{Fields per Frame})$
- RT is the time it takes to make one frame.
- Frame Rate is how many frames appear every second.
- Fields per Frame depends on the video standard; for instance, it's 2 for interlaced PAL.
- For example, if you're dealing with interlaced PAL video running at 25 frames per second, the RT would be calculated as:
- $RT = 1000 / (25 * 2) = 1000 / 50 = 20 \text{ milliseconds for one frame.}$
- So, in this case, it should take 20 milliseconds to create each frame for the video to look smooth. This metric helps ensure that frames are generated quickly enough for a smooth video experience.

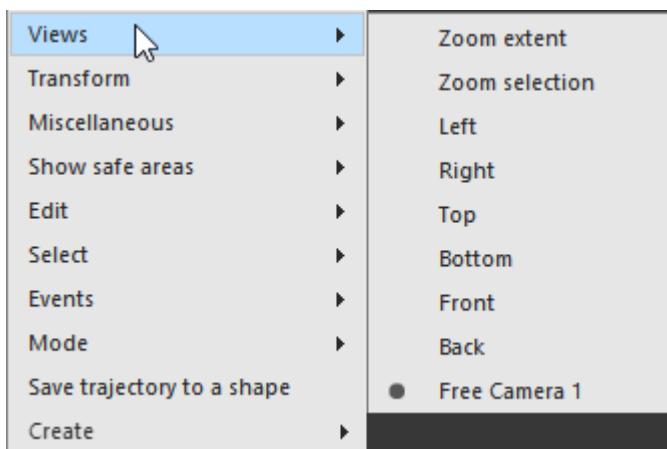
- RTS (Render Fluctuation Time): This value indicates the maximum fluctuation time experienced during rendering. A constant fluctuation may require you to make the scene lighter to avoid rendering issues.
- PR (Pre-Render Time): It displays the time taken to execute the pre-render event, which occurs before each frame. If this time exceeds 2 milliseconds, it is shown; otherwise, it displays 0.
- ST (System Time): This section shows the current date and time.
- WT (Working Time): It displays the time elapsed since Drone Designer was turned on. The time is shown in the format hh:mm:ss:ff (hours, minutes, seconds, and frames).

Lastly, when you right-click on the viewport, it provides a drop down menu with various options for your view-port.

Right click drops down options

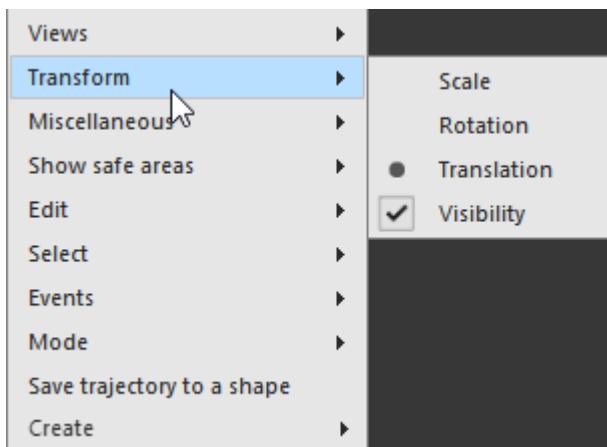
(To display set of options, right click on view-port) the drop down options are as follows:

• VIEWS



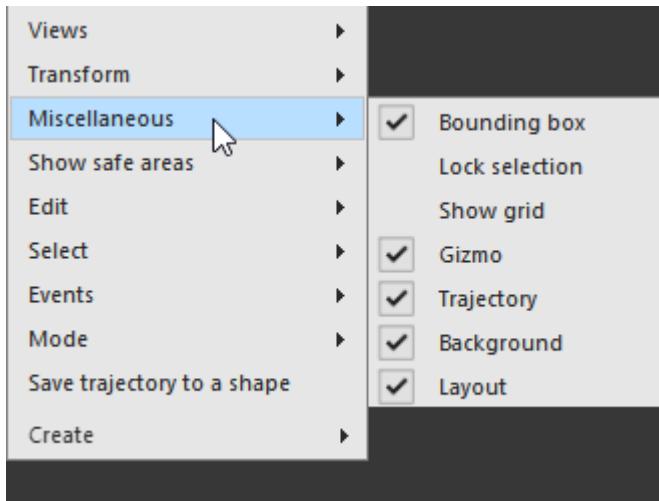
- Zoom Extent: This command fits the selected object to the screen.
- Zoom Selection: This command zooms in on the selected object.
- Left: This command switches the existing camera view to a left orthogonal view.
- Right: This command switches the existing camera view to a right orthogonal view.
- Top: This command switches the existing camera view to a top orthogonal view.
- Bottom: This command switches the existing camera view to a bottom orthogonal view.
- Front: This command switches the existing camera view to a front orthogonal view.
- Back: This command switches the existing camera view to a back orthogonal view.
- Free Camera 1: This command switches the existing camera view to a perspective free camera view. This is the default camera in the scene and is created by default. If there are other cameras in the scene, their names will be added to the Views list.

• TRANSFORM



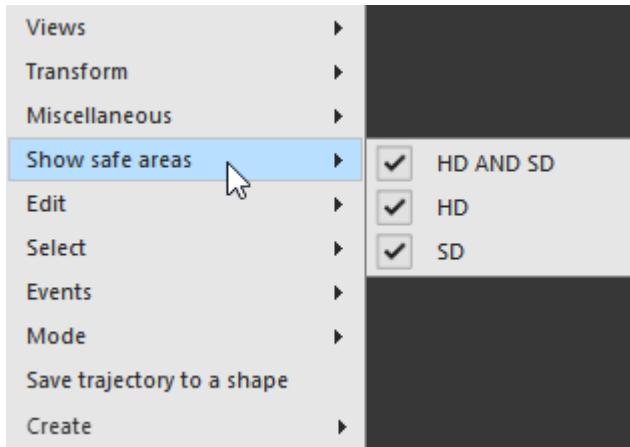
- Scale: Sets the tool selection to Scale
- Rotate: Sets the tool selection to Rotate.
- Translation: Sets the tool selection to Translation.
- Visibility - Toggles the visibility of the object.

- MISCELLANEOUS



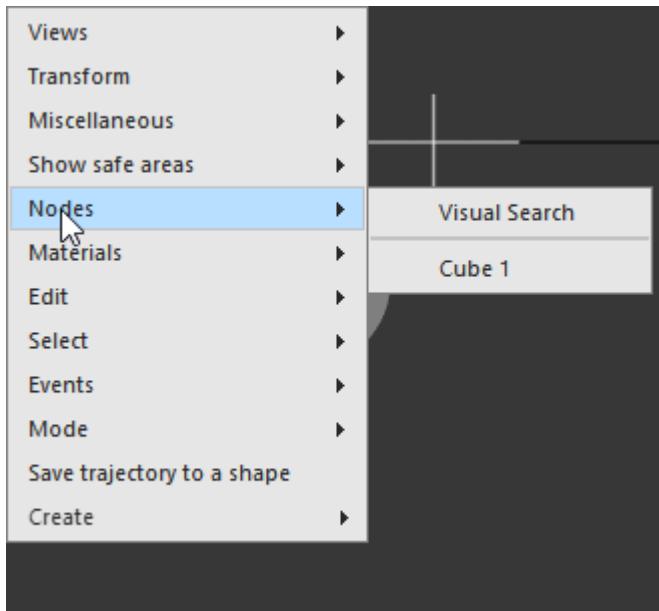
- Bounding Box: Toggles the display of the bounding box of the selected object.
- Lock Selection: Locks the selection of the selected object and does not allow any other object to be selected.
- Show Grid: Toggles the display of the grid on the viewport.
- Gizmo: Toggles the display of the transform Gizmo of the objects.
- Trajectory: Toggles the display of the animation trajectory and pivot of the scene objects.
- Background: Toggles the display of the background layer.
- Layout: Toggles the display of the layout layer.

- Show Safe Area - Shows the safe area frame on the viewport.

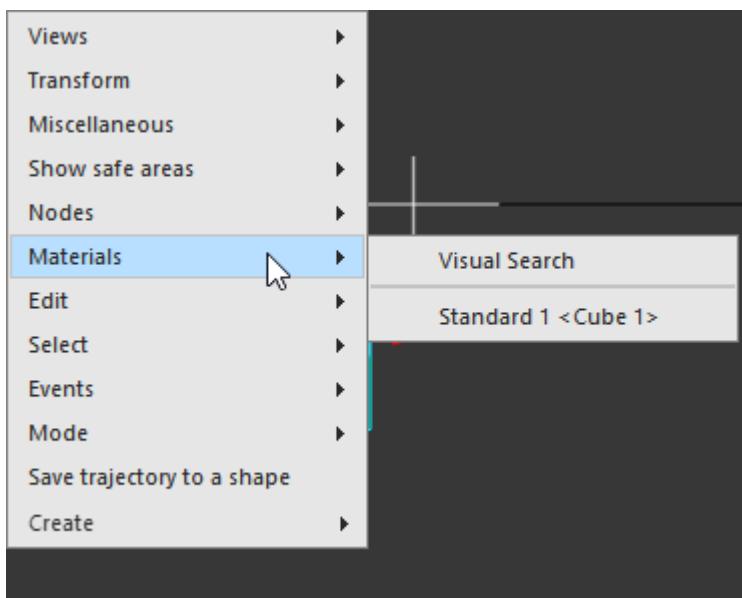


- HD: Displays the HD safe Area
- SD: Displays the SD safe Area

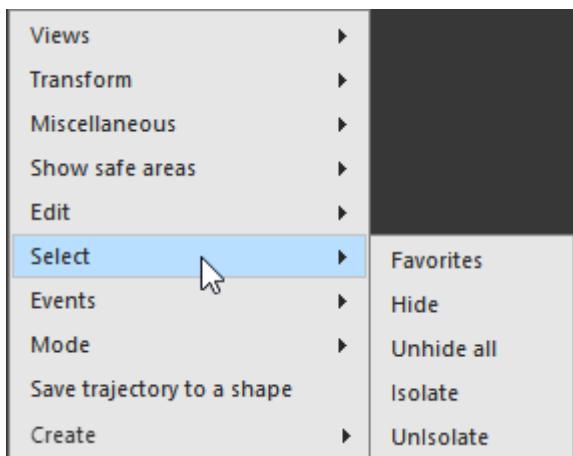
- Node



- Materials

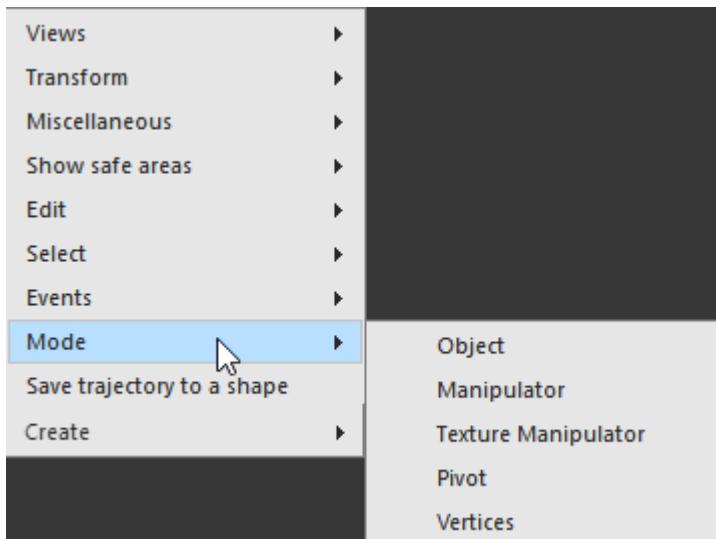


- Select



- Favorites: Opens the Add to Favorites dialog box for the selected object.
- Hide: Hides the selected object.
- Unhide: Unhides all the objects in the scene.
- **Isolate:** isolates the selected object in the scene, hiding all other objects. This can be helpful when you need to focus on a specific object without being distracted by other objects in the scene. Press "ALT+Q" as a shortcut for this command.
- Unisolate: Unisolate the selected object and restores the scene back with all objects visible.

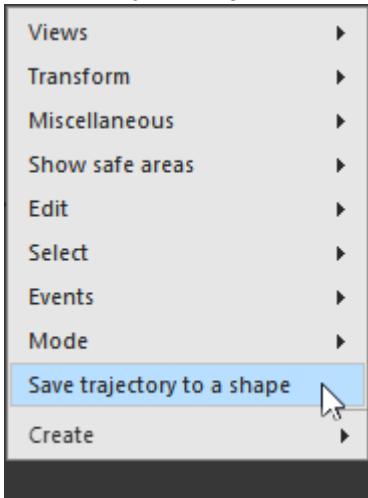
- Mode



- **Object (O):** This mode is the default mode of Drone Designer. The object transformations can be done in the viewport in this mode. This mode is activated by pressing the 'O' shortcut key when working in the Viewport.
- **Manipulator (M):** This mode provides additional helper gizmos to do parametric changes to the objects right from the viewport. This mode is activated by pressing the 'M' shortcut key when working in the Viewport.
- **Texture Manipulator(T):** This mode allows user to translate and scale the texture on the selected object. This mode is activated by pressing the 'T' shortcut key when working in the Viewport.
- **Pivot (I):** This mode allows user to translate the pivot in the viewport. Right-Click on the pivot mode icon to display the pivot transformation dialog box. This mode is activated by pressing the 'I' shortcut key when working in the Viewport.
- **Vertices (V):** This mode is only available for 2D objects. In this mode enables user to transform vertices of the selected 2D object. All changes done in the vertices mode are animatable. This mode is activated by pressing the 'V' shortcut key when working in the Viewport.
- **Polygon(G):** Polygon mode allows you to select and transform individual polygons of a 2D Object. This mode is only available for objects that contain multiple polygons, such as 2D text, shapes imported from Illustrator, or shapes that have been combined using the Composite Shape option in Designer. If you want to apply a different fill color to a specific polygon, you can press the ALT key while dragging a color from the color swatch onto the

polygon. This will apply the color to the selected polygon only, without affecting the other polygons in the object.

- Save trajectory to a shape



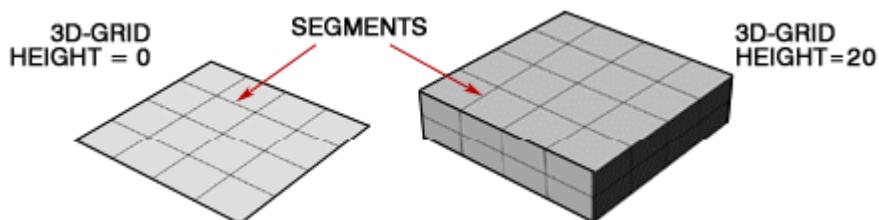
- Creates trajectory of an animated object to a shape.

3D Grid



Create > 3D Primitives > 3D Grid

Click on the **3D Grid**  tool icon in the 3D Primitives tool group.



Creating a 3D Grid

Click Create > 3D Primitives > 3D Grid from the menu.

Take your cursor to the Viewport and click and drag left and then up or down to draw rectangle of the grid.

Now, drag your mouse upward to set the depth of the grid. The point from where you will start will be center (Z depth) of the 3D Grid and define the radius of the 3D Grid.

Release the cursor to create the final shape.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Width	200.00
	Height	200.00
	Depth	200.00

1. Width: Defines width of a 3D Grid on the X-axis.
2. Height: Defines height of a 3D Grid on the Y-axis.
3. Depth: Defines depth of a 3D Grid on the Z-axis.

Extended Parameters

Extended Parameters		
	Width Segments	1
	Height Segments	1
	Depth Segments	1
	Show Lines	<input type="checkbox"/>
	Smoothing Angle	-1

1. Width Segments: Defines number of width segments. The width segments can be set between 1 to 50.
2. Height Segments: Defines number of height segments. The height segments can be set between 1 to 50.
3. Depth Segments: Defines the number of depth segments. The depth of the segments can be set between 1 to 50.
4. Show Lines: Select this check box to show the segments of 3D Grid. By default, it remains unchecked.
5. Smoothing Angle: Determines the number of segments between the two edges' vertices that define the curvature of the bevel edges.

Texture Options

Texture Options		
	Mapping Mode	PerFace

1. Mapping Mode: A texture must be applied to the object to see the Mapping Mode effect. It gives options to set Wrap or Per-Face mapping mode.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.

3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable placeholder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

3D Text

3D Text tool is one of the 3D Primitive tools and enables creating with native extrusion support. It supports multiple formats like free text, bounded text, clock, date, integer, decimal, countdown (time), count down (date), count down or stop watch, start time etc.



Create > 3D Primitives > Text 3D

Click on the  (3D Text tool) from the 3D Primitives tools group.

Parametric Attributes

Text Attributes

Text Attributes		
	Text	Press F11 to enter text
	Text Type	Free
	Extrude	0.00
	Inner Text	3D Text
	Numeric Text	0.00
	Format	CnclDefault
	Text Length	7
	Vertical Text	<input type="checkbox"/>

1. Text: Edit/modify text of the Text3D object in rich-text editor. Allows multiline text, press Enter after each line to start new line.
2. Text Type: Choose from a list of Text3D formats like Integer, Digital Clock, Countdown Clock, Stopwatch etc.
 - Free: It allows you to create a simple text object without 'Word Wrap' functionality.

- Time: Set this text type to create a digital clock object, which shows the system's time. After selecting this attribute, set the relevant parameter under Time/Date Options section.
- Date: Select this text type to create a date object and show system's date. After choosing the text type as 'Date', set the correct format under Time/Date Options section.
- Numbers- Use this text type to create an Integer or a Float numeric Text, depending on the usage.
 - At default state, the numbers are defined as an Integer.
 - When the numbers are changed by pressing Control key, the numbers change in a Decimal value.
- Countdown: This Text type allows you to create a countdown for Time and Date.

3. **Extrude:** Adds depth to the text.

4. Inner Text: Edit/modify text of the Text3D object in rich-text editor. Allows multiline text, press Enter after each line to start new line.
5. Numeric Text: Sets the numerical value of 3d Text as an Integer or Decimal Type.
6. Format: Specify format of the Clock/Date/Countdown/Stopwatch as per Windows Date and Time format method.
7. Text Length: Shows the number of Characters in the text including space.
8. Vertical Text: Toggles vertical text.

Font Attributes

Font Attributes		
	Font Name	Arial
	Size	60
	Weight	Normal
	Italics	<input type="checkbox"/>
	Underline	<input type="checkbox"/>
	Tracking	0.00
	Space Multiplier	1.00
➤	Fore Color	255,255,255,255
	Steps	3

1. Font Name: Choose from a list of all available Windows fonts(.TTF and .OTF)
2. Size: Sets the font size of the text. The size measurement method depends on the font.
3. Weight: Sets the font Weight or boldness(Bold, Black, Thin etc.) of the text. Only weights supported by font will be displayed.
4. Italics: Toggles italicized text.
5. **Underline:** Toggles Underlined text.
6. **Tracking:** Sets the distance between letters.
7. **Space Multiplier:** Multiplies the spacing between characters of a text string. Best used in Korean and Chinese fonts where multiple glyphs are used to create a word on same location.
8. Fore Color: Applies the flat base color of the text. It sets the initial color of the RTF text. If no material is applied, this is the color in output.
9. Steps: Specifies the number of divisions between each edge of the Text.

Alignment Attributes

Alignment Attributes		
	Horizontal Alignment	Centre
	Vertical Alignment	Centre
	Line Spacing	0.00
	Base Embedding	Left to Right

1. Horizontal Alignment: Sets the horizontal orientation of the Text based on the Pivot or Bounding box of the text.
2. Vertical Alignment: Sets the vertical orientation of the Text based on the Pivot or Bounding box of the text.
3. Line Spacing: Adjusts the distance between lines of text. Effect only visible when text has multiple lines.
4. Base Embedding: Sets the Text embedding method. Choose Right to Left for Arabic.

Bounded Text Options

Bounded Text Options		
	Bounded Text	<input type="checkbox"/>
	Show Bounded Rect	<input checked="" type="checkbox"/>
	Action on Overflow	Fit
	Bounded Width	600.00
	Bounded Height	200.00
	Fit Threshold	0
	Scroll Time	4
	Reveal Delay	100
	Reveal Time Factor	100

1. Bounded Text: Toggles text wrapping area constraints.
2. Show Bounded Rect: Toggles display of Bounded Box(Purple Color) of the Bounded type of Text.
3. Action on Overflow: Sets the text adjustment method in-case the amount of text exceeds the bounding area.
4. Bounded Width: Specifies the Width of the Bounding Box of Bounded type Text.
5. Bounded Height: Specifies the Height of the Bounding Box of Bounded type Text.
6. Fit Threshold: Specifies the percentage of maximum length allowed to fit.
7. Scroll Time: Specifies the time taken to scroll text.
8. **Reveal Delay:** Specifies the delay to start revealing the text in percentage of line height scrolled.
9. **Reveal Time Factor:** Specifies the time taken to reveal the text.

Text Animation

Text Animation		
	Character Effect	1.0000,0.0000,0.0000,0.00...
	Animation Texture	None

1. Character Effect: Animate the characters of the text individually using the interpolation of the SRTV key frames.
2. Animation Texture: Used with Texture-Wise character animation. The characters animate according to the Luminance (darkest region to lightest) of the selected texture.

Extended Parameters

Extended Parameters		
	Spline	None
	Countdown Target	12/22/2022 2:51 PM
>	Stopwatch Counter	0.00.0.00,10.00.0.00
	Time Offset	<input type="checkbox"/>
	Ignore White-Space	<input type="checkbox"/>
	Material Color Front	<input checked="" type="checkbox"/>
	Material Color Walls	<input checked="" type="checkbox"/>
	Material Color Back	<input checked="" type="checkbox"/>
	Digit Substitution	None
	Detect Winding Order	<input type="checkbox"/>

1. Spline: Select the shape object to constrain text movement along its spline.
2. Countdown Target: Specifies the date/time up to which the Countdown will run.
3. Stopwatch Counter: Specifies the Counter value in Positive for Count-Up and Negative for Countdown of the Stopwatch type Text.
4. **Time Offset:** Specifies an offset value to the Specified Time or Stopwatch type Text separated by colon e.g. h+2:m+30.
5. Ignore White-Space: Toggle the visibility of space as empty or with box in E-Text object or highlighted text.
6. Material Color Front: When checked, material color will override the color specified on front face of Rich Text.
7. Material Color Walls: When checked, material color will override the color specified on side faces of Rich Text.
8. Material Color Back: When checked, material color will override the color specified on back face of Rich Text.
9. Digit Substitution: Sets the Digit Substitution method.
10. Detect Winding Order: Auto detect the character(polygon) winding order.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Candle Graph

A combination of "line-chart" and "bar-chart" that can be used to show a range of price movement over a given time interval.



Create > 3D Primitives > Candle Graph

Click on the  (Candle Graph) from the 3D Primitives tool group.

Default		
	Element Animation	1.0000,0.0000,0.0000,0.0000
	Candle Object	None
	Stick Ratio	10.00
	Gap Ratio	50.00
	Axis of Orientation	Y
	Graph Width	400.00
	Graph Height	400.00
	Plot Data	0.0 0.5 0.4 0.7 0.1 0.25 0.3 0...
>	Up Candle Color	 0,255,0,255
>	Down Candle Color	 255,0,0,255
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

Default

1. Element Animation - Effect applied on Candle Graph.
2. Candle Object - Specifies the reference object to be used as candles for Candle graph. Best works with Objects which have height in their parameters for e.g., Cylinder, Cube, N-Gon etc.
3. Stick Ratio - Sets the width of the stick of each Candle
4. Gap ratio - Sets the gap between two Candles
5. Axis of Orientation - Sets the Orientation of the reference object used as candle. By default, none takes the default orientation of the reference object.
6. Graph Width - Sets the size in X-axis
7. Graph Height - Sets the size in Y-axis.
8. Plot Data - Provides and interface to enter the historical Graph Values(Open, High, Low, Close). Optionally this parameter can be wired with the FinWiz data control to get data from the Financial Feed.
9. Up Candle Color - Sets the color of the candles showing positive graph values.
10. Down Candle Color - Sets the color of the candles showing negative graph values.
11. Candlestick Effect - Effect applied on Candle Graph
12. Texture Modifier: Specifies the Planar Texture Mapping object for planar UVW texture mapping on the object.
13. **Rotation Order:** Allows user to specify the required Rotation Order. It helps user to assign the required order of Rotation for animating the object as per requirement.

14. Tag: Serves as a variable place holder to hold values for the object. User can specify object names in Tag and wire the equivalent properties of the objects by specifying Object.Tag as Target.

Cloth

This is a 3D Primitive tool. This tool is very useful for creating cloth animation and giving it special effects using various attributes such as Gravity, Wind, Guide etc.



Create > 3D Primitives > Cloth

Click on the  (Cloth tool) from the 3D Primitives tool group.

Parameters Attributes

Standard Parameters

Standard Parameters		
	Width	400.00
	Height	200.00
	Segments (Horz)	16
	Segments (Vert)	8
	Alignment	Centre

1. Width: Sets the size along the X-axis
2. Height: Sets the size along the Y-axis.
3. Segments (Horz): Specifies the number of segments along width of the cloth. Results in more natural cloth animation.
4. Segments (Vert): Specifies the number of segments along height of the cloth. Results in more natural cloth animation.
5. Alignment: Choose the pivot position of the cloth.
 - Center: Aligns the center of cloth object to the origin of the trajectory.
 - Top Left: Aligns the top left side of cloth object to the origin of the trajectory.
 - Top Right: Aligns the top right side of the cloth object to the origin of the trajectory.
 - Bottom Left: Aligns the bottom left side of the cloth object to the origin of the trajectory.
 - Bottom Right: Aligns the bottom right side of the cloth object to the origin of the trajectory.
 - **Center Left:** Aligns the center left side of the cloth object to the origin of the trajectory.
 - **Center Right:** Aligns the center right side of the cloth object to the origin of the trajectory.

Extended Parameters

Extended Parameters		
	Select Spline Extrude	None
	Pin Spline	<input checked="" type="checkbox"/>
	Pin Start Cap	<input checked="" type="checkbox"/>
	Pin End Cap	<input checked="" type="checkbox"/>
	Guide	None

1. Select Spline Extrude: Select the Spline Extrude Object from the scene to transform cloth to the shape of spline extrude.
2. Pin Spline: Pins the cloth animation along the Spline of the Spline Extrude. Works only when a spline extrude object is selected.
3. Pin Start Cap: Pins the cloth animation along the Start Cap of the Spline Extrude. Works only when a spline extrude object is selected.
4. Pin End Cap: Pins the cloth animation along the End Cap of the Spline Extrude. Works only when a spline extrude object is selected.
5. Guide: When Specified, the cloth follows the selected object and moves according to the selected object's animation and orientation.

Cloth Dynamics

Cloth Dynamics		
	External Forces	None
>	Gravity	0.00,-3.00,0.00
>	Wind	15.00,2.00,2.00

1. External Forces: Specifies the External Gravity or Wind force to be applied to the cloth. Works in addition to the native gravity and wind.
2. Gravity: Simulates the effect of gravity(in X,Y,Z direction) on cloth. Results in stretching of the cloth.
3. Wind: Simulates the effect of Wind (in X,Y,Z direction) on Cloth. Results in fluttering of the cloth.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Cube

This is one of the 3D Primitive tools and can be used to create cube, dice, and 3D rectangle. By default comprises of chamfer, chamfer segments, and smoothing angle value as 1.0000, 1, and -1 respectively. Value of other parameters such as, width, height, and depth depends on whether you are creating a cube manually or by double-clicking the Cube Tool.



Create > 3D Primitives > Cube

Click on the  (**Cube tool**) icon in the 3D Primitives tool group

Following are the two profiles that can be changed from Standard Parameters section of Parameters (F4) window:

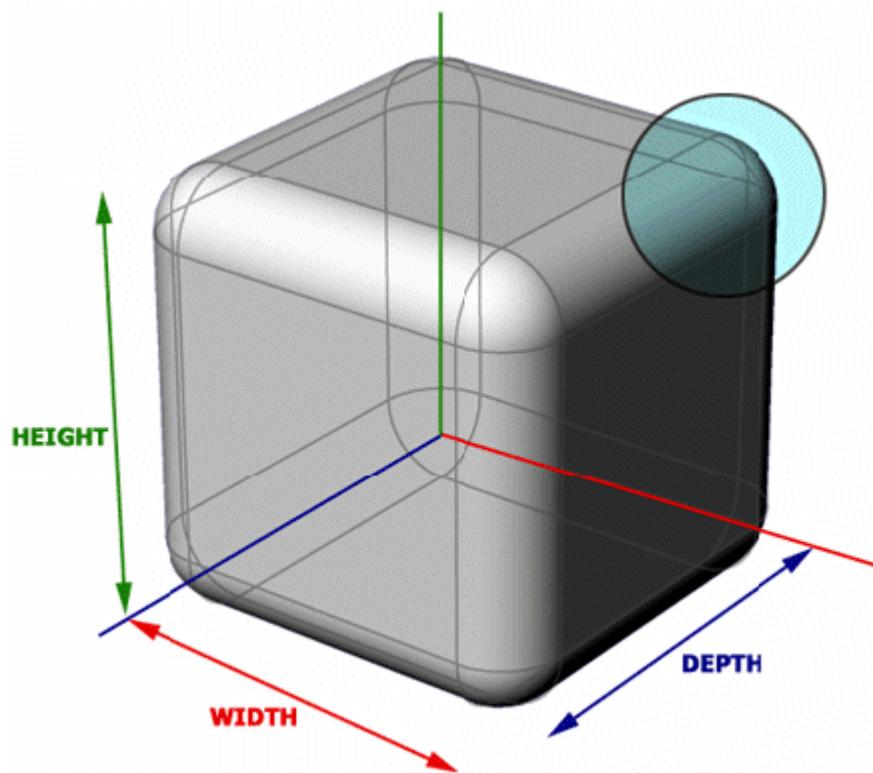
1. Cube: Lets you create a cube. This is the default value for a cube.
2. Dice: Lets you create a dice shape.

Creating a Cube

1. Click Create > 3D Primitives > Cube from the menu.
2. Click and drag your mouse on the viewport to draw a cube. The point from where you will draw the cube will be the center (Z depth) of the cube and define the height/width/depth of the cube.
3. Release the cursor to create the final shape. This is the default shape.

Creating a Dice

1. Click Create > 3D Primitives > Cube from the menu.
2. Click and drag your mouse on the viewport to draw a cube. The point from where you will draw the dice will be the center (Z depth) of the cube and define the height/width/depth of the dice.
3. Release the cursor to create the final shape. This is the default shape.
4. From the Parameters window, change the Profile Property to Dice.



Parametric Attributes

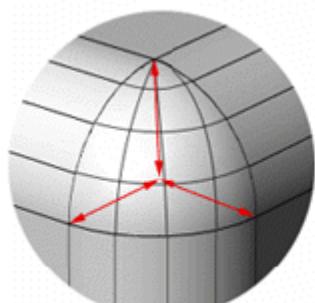
Standard Parameters

Standard Parameters		
	Width	200.00
	Height	200.00
	Depth	200.00
	Base Pivot	Centre

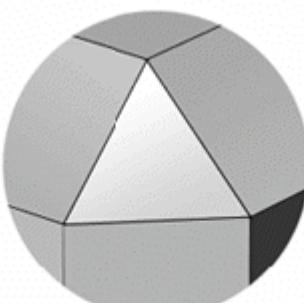
1. Width: Sets the size along the X-axis.
2. Height: Sets the size along the Y-axis.
3. Depth: Sets the size along the Z-axis.
4. Base Pivot: Shifts the origin of the Cube. The Height parameter works according to Alignment.

Extended Parameters

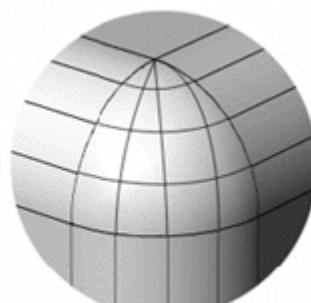
Extended Parameters		
	Profile	Cube
	Chamfer Radius	0.00
	Chamfer Segments	1
	Smoothing Angle	-1



Chamfer (Edge Distance)



Chamfer Segment = 1



Chamfer Segment = 4

1. Profile: Specify the Chamfering method as Cube (all corners and edges) or Dice (all corners but only top and bottom edges).
2. Chamfer Radius: Sets the chamfer radius at sides and corners as per profile.
3. Chamfer Segments: Sets the number of segments in the chamfered edges.
4. Smoothing Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing.

Texture Options

Texture Options		
	Mapping Mode	Per Face

1. Mapping Mode: Sets the Texture Mapping method of Cube.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable placeholder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

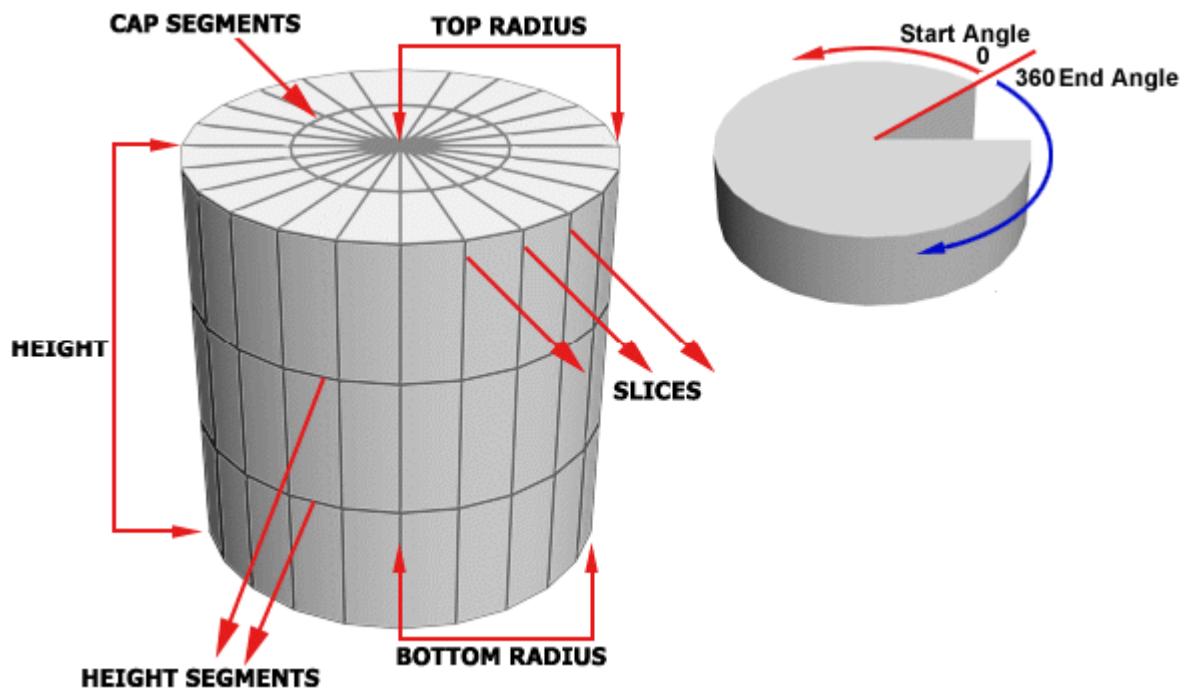
Cylinder

Cylinder is a 3D Primitive tool. It enables creating cylinder, which can also be manipulated to create conical forms.



Create > 3D Primitives > Cylinder

Click on the Cylinder Tool from the 3D Primitives tool group.



Parametric Attributes

Standard Parameters

Standard Parameters		
	Radius (Top)	100.00
	Radius (Bottom)	100.00
	Height	200.00
	Segments (Radial)	20
	Segments (Height)	1
	Start Angle	0.00
	End Angle	360.00
	Angle	360.00
	Base Pivot	Bottom

1. Radius (Top): Sets the Radius of the top side.
2. Radius (Bottom): Sets the Radius of the bottom side.
3. Height: Sets the size along the Y-axis
4. Segments (Radial): Sets the number of segment divisions or sides.
5. Segments (Height): Sets the number of Segments along the Height.
6. Start Angle: Sets the Angle around the axis from where the Slice starts.
7. End Angle: Sets the Angle around Y-axis from where the Slice ends.
8. Angle: Angle around Y-axis to define the total angle extents.

Extended Parameters

Extended Parameters		
	Chamfer Radius	0.00
	Chamfer Segments	1
	Smoothing Angle	-1

1. Chamfer Radius: Sets the Chamfer radius at Top and Bottom edges.
2. Chamfer Segments: Sets the segment divisions between the chamfered edges.
3. Smoothing Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing.

Capping Options

Capping Options		
	Top Cap	<input checked="" type="checkbox"/>
	Bottom Cap	<input checked="" type="checkbox"/>
	Start Cap	<input checked="" type="checkbox"/>
	End Cap	<input checked="" type="checkbox"/>

1. Top Cap: Toggles display of Top Cap of the Cylinder.
2. Bottom Cap: Toggles display of Bottom Cap of the Cylinder.
3. Start Cap: Toggles display of Start Cap of the sliced Cylinder.
4. End Cap: Toggles display of End Cap of the sliced Cylinder.

Texture Options

Texture Options		
	Cap Texture	<input checked="" type="checkbox"/>
	Wall Texture	<input checked="" type="checkbox"/>
>	Cap UV Multiplier	1.00,1.00
>	Wall UV Multiplier	1.00,1.00

1. Cap Texture: Toggles visibility of texture of the applied Material on Top and Bottom faces.
2. Wall Texture: Toggles visibility of texture of the applied Material on Start and End Caps of the slice.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.

3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Layer

Use Layer to create a 2D orthographic plane object. It does not have any width or height options. it is always the size of the render view.



Create > 3D Primitives > Layer

Click on the  (Layer tool) from the 3D Primitives tool group.

Parametric Attributes

Standard Parameters

Standard Parameters	
Kill Z-Depth	<input type="checkbox"/>
Source Texture	None
Fit Options	Fit
Object Type	Layer
Pass to Texture	<input type="checkbox"/>

1. Kill Z-Depth: Toggles the rendering of the objects, which are behind the Layer object in Z-Depth. When On Layer Will behave as a background in the specified Z depth before camera.
2. **Source Texture:** Specify the texture to be applied to the Layer from the local drive.
3. Fit Options: Choose from a list of Fit options like Letterbox , Fill etc.
4. Object Type: Choose from a list of Object Type like Layer, Background etc.
5. Pass to Texture: Transfer the call to Texture.

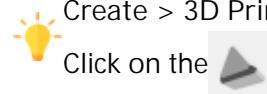
Default

Default	
<input type="checkbox"/>	Tag

1. Tag: Serves as a variable place holder to hold values for the object. User can specify object names in Tag and wire the equivalent properties of the objects by specifying Object.Tag as Target.

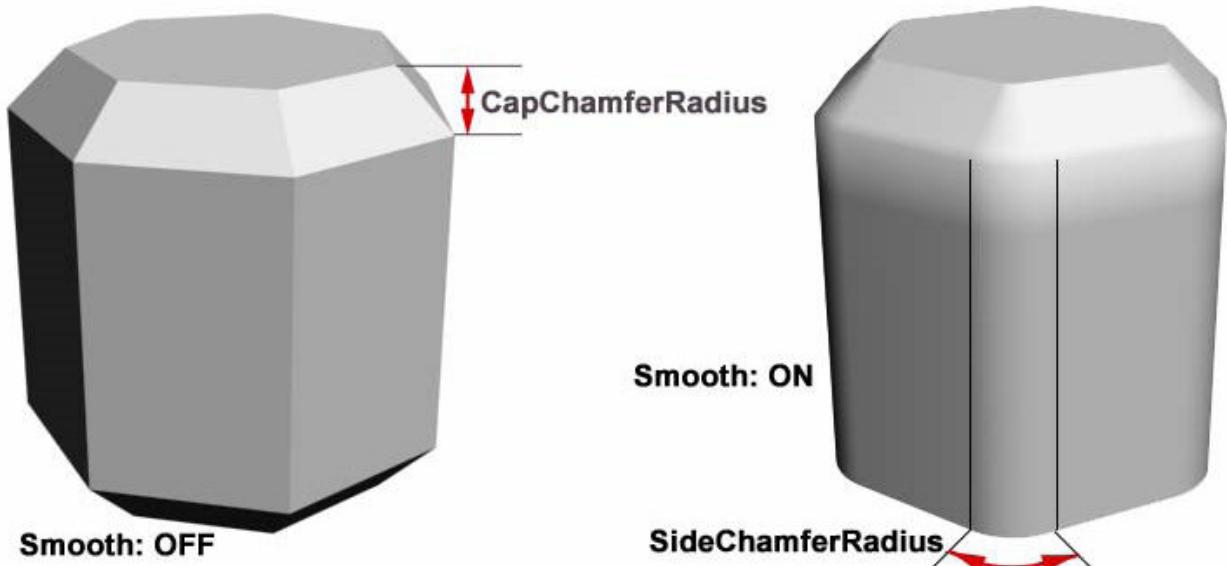
N_Gon

This is one of the 3D Primitives tools, which enables creating N-sided 3D polygonal mesh. In N-Gon, the N represents the number of sides in a polygon.



Create > 3D Primitives > N-Gon

Click on the  (N-Gon tool) from the 3D Primitives tool group.



Parametric Attributes

Standard Parameters

Standard Parameters		
	Sides	3
	Radius	120.00
	Height	200.00
	Alignment	Bottom

1. Sides: Sets the number of sides
2. Radius: Sets the Radius.
3. Height: Sets the size along the Y-axis.
4. Alignment: Shift Origin.

Extended Parameters

Extended Parameters		
	Chamfer Cap Radius	1.00
	Chamfer Cap Segments	0
	Chamfer Walls Radius	1.00
	Chamfer Wall Segments	1
	Smoothing Angle	-1

1. Chamfer Cap Radius: Specifies the Chamfer Radius on the Top and Bottom edges.
2. Chamfer Cap Segments: Sets the number of segments in the top and bottom chamfered edges.
3. Chamfer Walls Radius: Specifies the Chamfer Radius on the side edges.

4. Chamfer Wall Segments: Sets the number of segments in the side chamfered edges.
5. Smooth Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing.

Texture Options

Texture Options		
	> Cap UV Multiplier	1.00,1.00
	> Wall UV Multiplier	1.00,1.00

1. Cap UV Multiplier: Specifies Cap Multiplier.
2. Wall UV Multiplier: Specifies Wall Multiplier.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable placeholder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

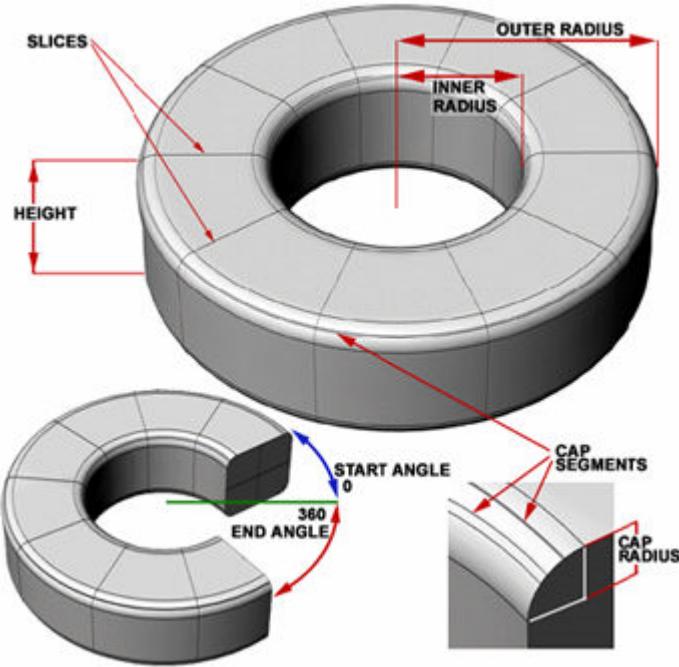
Ring

Ring is a 3D Primitive tool, which enables creating ring. Its best use is in the form of a Pie Graph.



Create > 3D Primitives > Ring

Click on the  Ring tool from the 3D Primitives tool group.



Parametric Attributes

Standard Parameters

Standard Parameters		
	Radius (Inner)	75.00
	Radius (Outer)	200.00
	Height	50.00
	Segments	40
	Start Angle	0.00
	End Angle	360.00
	Angle	360.00
	Base Pivot	Bottom

1. Radius (Inner): Sets the Hole radius.
2. Radius (Outer): Sets the outer radius.
3. Height: Sets the size along the Y-axis
4. Segments: Sets the number of segment divisions or sides.
5. Start Angle: Sets the Angle around Y-axis from where the Slice starts.
6. End Angle: Sets the Angle around Y-axis from where the Slice ends.
7. Angle: Angle around Y-axis to define the total angle extents.
8. Base Pivot: Shifts the origin of the Ring.

Extended Parameters

Extended Parameters		
	Chamfer Radius	6.00
	Chamfer Segments	4
	Smoothing Angle	-1

1. Chamfer Radius: Chamfers the inner and outer edges.
2. Chamfer Segments: Sets the segment divisions between the chamfered edges.

3. Smoothing Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing.

Capping Options

Capping Options		
	Start Cap	<input checked="" type="checkbox"/>
	End Cap	<input checked="" type="checkbox"/>

1. Start Cap: Toggles display of Start Cap of the slice.
2. End Cap: Toggles display of End Cap of the slice.

Texture Options

Texture Options		
	Mapping Mode	Wrap
>	UV Multiplier	1.00,1.00

1. Mapping Mode: Set the texture mapping method of Ring.
2. UV Cap: Specifies UV multiplier.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies weather the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

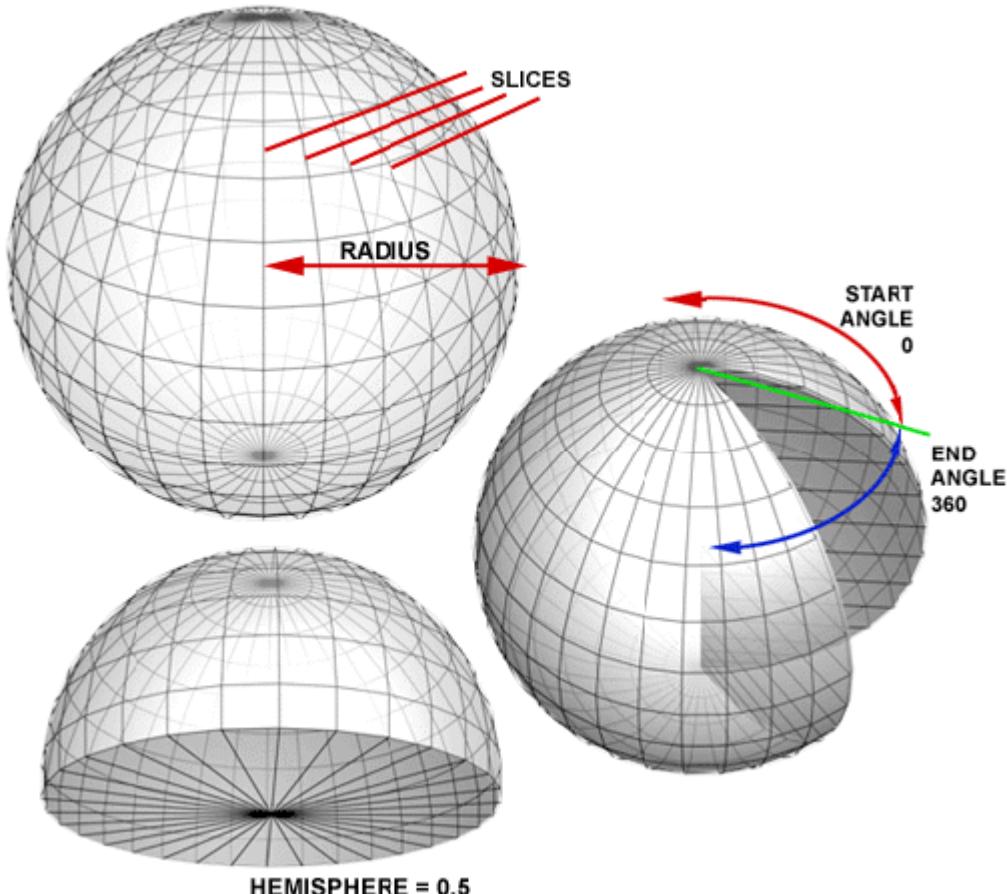
Sphere

Sphere is a 3-dimensional shape like a ball in which every point on the surface is at the same distance from the center. This tool is a part of the 3D Primitive tools group, which enables you to create a sphere, hemisphere, or a free form slice.



Create > 3D Primitives > Sphere

Click on the (Sphere icon) in the 3D Primitives tool group



Creating a Sphere

1. Click Create > 3D Primitives > Sphere from the menu.
2. Take your cursor to the Viewport and click and drag to draw a sphere. The point from where you will draw the sphere will be the center (Z depth) of the sphere and define the radius of the sphere.
3. Release the cursor to create the final shape.

Creating a Hemisphere

1. Click Create > 3D Primitives > Sphere from the menu.
2. Take your cursor to the Viewport and click and drag to draw a sphere. The point from where you will start will be the center (Z depth) of the sphere and define the radius of the sphere.
3. Release the cursor to create the final shape.
4. Change the Chamfer Top from 0 to 50 to cut the slice from top. Alternatively, you can change the chamfer bottom from 100 to 50 to cut the sphere from bottom.

Parametric Attributes

Standard Parameters

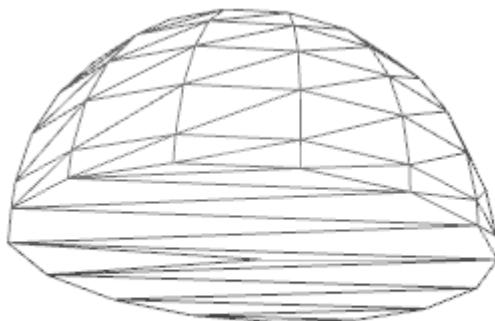
Standard Parameters		
	Radius	100.00
	Segments	20
	Start Angle	0.00
	End Angle	360.00
	Angle	360.00

1. Radius: Sets the radius.
2. Segments: Sets the number of segment divisions.
3. Start Angle: Sets the Angle around Y-axis from where the Slice starts.
4. End Angle: Sets the Angle around Y-axis from where the Slice ends.
5. Angle: Angle around Y-axis to define the total angle extents.

Extended Parameters

Extended Parameters		
	Chamfer Top (%)	0
	Chamfer Bottom (%)	100
	Chamfer Offset (%)	0
	Smoothing Angle	-1

1. Chamfer Top (%): Sets the Top Chamfer, results in Chopping from the top end along Y-axis.
2. Chamfer Bottom (%): Sets the Bottom Chamfer, results in Chopping from the bottom end along Y-axis.



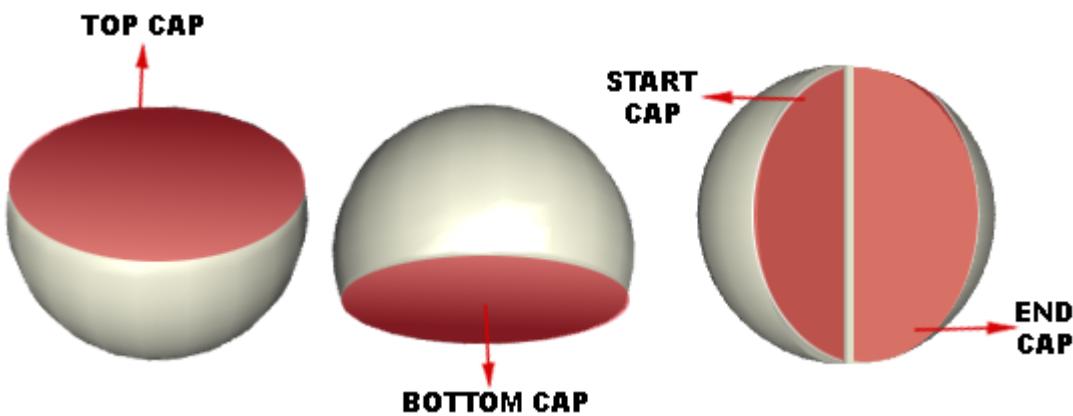
Chamfer Bottom = 50

3. Chamfer Offset (%): Offsets between the Top and Bottom chamfer of Sphere.
4. Smoothing Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing.

Capping Options

Capping Options		
	Top Cap	<input checked="" type="checkbox"/>
	Bottom Cap	<input checked="" type="checkbox"/>
	Start Cap	<input checked="" type="checkbox"/>
	End Cap	<input checked="" type="checkbox"/>

The Capping Options can be viewed when Chamfer Parameters are set.



1. Top Cap: Toggles display of Top Cap of the top chamfered Sphere.
2. Bottom Cap: Toggles display of Bottom Cap of the bottom chamfered Sphere
3. Start Cap: Toggles display of Start Cap of the slice.
4. End Cap: Toggles display of End Cap of the slice.

Texture Options

Texture Options	
Cap Texture	<input checked="" type="checkbox"/>
Wall Texture	<input checked="" type="checkbox"/>

1. Cap Texture: Toggles visibility of texture of the applied Material on Top and Bottom faces of the chamfered Sphere.
2. Wall Texture: Toggles visibility of texture of the applied Material on Start and End Caps of the slice.

Instancing Options

Instancing Options		
Target	None	
Range	-1	
Wire Animation	<input type="checkbox"/>	

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
Texture Modifier	None	
Rotation Order	XYZ	
Tag		

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allows user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable placeholder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Import Skeletal Mesh

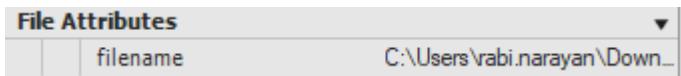
This is an extension tool that can be used to create a 3D beveled mesh from a 2D shape. A bevel object extrudes a shape into 3D object and applies a flat or smooth bevel on the edges of the object.



Create > Biped Kinematics > Skeletal Mesh

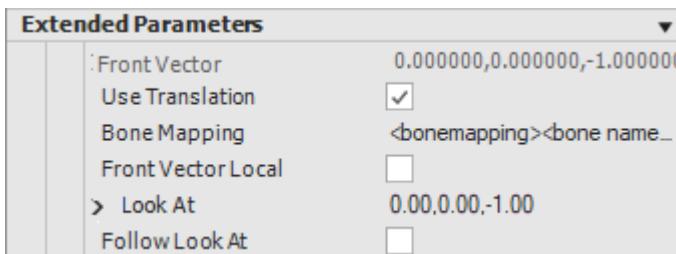
Click on the  (Skeletal Mesh) from the **Biped Kinematics** tool group.

File Attributes



Filename: Displays the file path of the FBX file or enables to change the file.

Extended Parameters



Front Vector: Front Direction Vector

Use Translation: When enabled, uses translation data from FBX file else translation is not used from the FBX and the model would stay on its location.

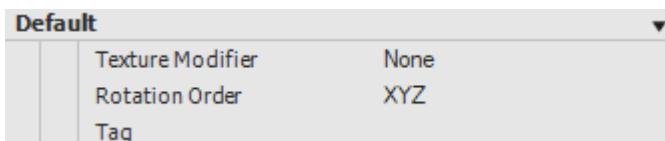
Bone Mapping: Mapping of skeletal bones with their animation counter parts

Front Vector Local: Whether to transform front vector with local matrix or global.

Look At: Allows user to specify a location so that the skeletal front is faced towards this position.

Follow Look At: When enabled, the Skeletal object follows the Look-At position.

Default



1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Directional

This is one of the Camera and Lights tools. When Gesture mode is enabled with Drone Designer's Viewport, it allows designers to test interactivity between different scene objects when creating touch-based graphics.

Using this tool, you can create a directional light in the scene, which has material A material is a design that you assign to the surface or face of an object so that it appears a certain way when rendered. Materials affect the color of objects, their glossiness, their opacity, and texture. based effect, i.e. different materials can have different light sources.



Create > Camera and Lights > Directional Light



Click on the (Directional Light Tool) from the Camera and Lights tool group.

Standard Parameters

Standard Parameters		
	Light Color	255,255,255,255
	Pixel Sampling count	32
	Blocker Sample Count	16.00
	Voxel Angle	2.40
	Light Size	0.25
	PenumMax	0.02

Light Color: Sets the color for the light.

Pixel Sampling Count: Displays the number of pixels to be sampled for shadow.

Blocker Sample Count: Displays the number of pixels to be sampled in blocker area.

Voxel Angle: Sets the angle for generating voxel disk points.

Light Size: Sets the size of light source for shadow.

PenumMax: Sets the maximum penumbra region.

Default

Default		
	Rotation Order	XYZ
	Tag	

- Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
- Tag:** Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Free Camera

Free Camera focuses and captures the area where it is aimed. Unlike target camera, it does not have two controls for the target and camera respectively. A single icon represents free camera, making it easier to animate. Target view can be set in any direction.



Create > Cameras and Lights > Free Camera

Click on the  (Free Camera) icon in the Cameras and Lights tool group.

Standard Parameters

Default		
	Field of View	45.00
	Near Clip	1.00
	Far Clip	10,000.00
	Aspect Ratio	1.78
	View Type	Perspective
	Offset X	0.00
	Offset Y	0.00
	Camera FOV	<input checked="" type="checkbox"/>
	Camera Offset	5.00
	Override Aspect Ratio	<input type="checkbox"/>
	Override Ortho Camera Di...	<input type="checkbox"/>
	Ortho Width	0.00
	Ortho Height	0.00
	Rotation Order	XYZ
	Tag	

1. **Field of View:** Specifies how wide an area the camera views.
2. **Near Clip:** Specifies the near clip distance, the object closer than the near clip distance are not visible to camera.
3. **Far Clip:** Specifies the far clip distance, the object farther than the far clip distance are not visible to camera.
4. **Aspect Ratio:** Specifies custom aspect ratio for the camera when Override Aspect Ratio is turned ON. By default, aspect ratio is taken as per the Viewport size.
5. **View Type:** Sets the camera Type to Perspective or Orthogonal.
6. **Offset X:** Adds an offset distance in horizontal axis of the camera.
7. Offset Y: Adds an offset distance in vertical axis of the camera.
8. **Camera FOV:** Enables FOV based camera view
9. Camera Offset: Specifies the offset of principal Axis of the camera. Works when FOV camera is turned ON.
10. **Override Aspect Ratio:** When ON, applies the aspect ratio specified in the Camera Aspect Ratio
11. Override Back Buffer: Toggle width and height of engine as per orthographic camera.
12. Ortho Width: Specifies the screen width for orthographic camera.
13. Ortho Height: Specifies the screen height for orthographic camera.

14. **Rotation Order:** Allows user to specify the required Rotation Order. It helps user to assign the required order of Rotation for animating the object as per requirement.
15. **Tag:** Serves as a variable place holder to hold values for the object. User can specify object names in Tag and wire the equivalent properties of the objects by specifying Object.Tag as Target.

Target Camera

The Camera Viewport simulates the classic pinhole camera. It displays the three-dimensional scene (which is projected in perspective) to create a two-dimensional display. In other words, target camera is an entity that resides in three-dimensional space, and provides the origin of a view-frustum (eye-point) and is a pyramid encompassing what is visible in the world.



Create > Camera and Lights > Target Camera

Click on the  (Target Camera Tool) from the Camera and Lights tools group.

Standard Parameters

Standard Parameters		
	Field of View	45.00
	Near Clip	1.00
	Far Clip	10,000.00
	Aspect Ratio	1.00
	View Type	Perspective
	Offset X	-0.63
	Offset Y	-0.50
	Camera FOV	<input checked="" type="checkbox"/>
	Camera Offset	5.00
	Override Aspect Ratio	<input type="checkbox"/>

1. **Field of View:** Specifies how wide an area the camera views.
2. **Near Clip:** Specifies the near clip distance, the object closer than the near clip distance are not visible to camera.
3. **Far Clip:** Specifies the far clip distance, the object farther than the far clip distance are not visible to camera.
4. **Aspect Ratio:** Specifies custom aspect ratio for the camera when Override Aspect Ratio is turned ON. By default, aspect ratio is taken as per the Viewport size.
5. **View Type:** Sets the camera Type to Perspective or Orthogonal.
6. **Offset X:** Adds an offset distance in horizontal axis of the camera.
7. **Offset Y:** Adds an offset distance in vertical axis of the camera.
8. **Camera FOV:** Enables FOV based camera view
9. **Camera Offset:** Specifies the offset of principal Axis of the camera. works when FOV camera is turned ON.
10. **Override Aspect Ratio:** When ON, applies the aspect ratio specified in the Camera Aspect Ratio

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Virtual Camera



Create > Cameras and Lights > Virtual Camera

Click on the  (Virtual Camera) icon in the Cameras and Lights tool group.

Standard Parameters

Standard Parameters		
	Field of View	45.00
	Near	1.00
	Far	10,000.00
	Aspect Ratio	1.00
	View Type	Perspective
	Offset X	-0.63
	Offset Y	-0.50
	Camera FOV	<input checked="" type="checkbox"/>
>	FOV XY	0.00,0.00
>	Principal Axis XY	0.00,0.00
	Flag FOV	0
	Addin Aspect Ratio	0.00
	Override Aspect Ratio	<input type="checkbox"/>
	Conversion	<input type="checkbox"/>
	DataTableRowIndex	0

1. Field of View: Specifies how wide an area the camera views.
2. Near: Specifies the near clip distance, the object closer than the near clip distance are not visible to camera.
3. Far: Specifies the Far clip distance, the object farther than the far clip distance are not visible to camera.
4. Aspect Ratio: Specifies custom aspect ratio for the camera when override Aspect Ratio is turned ON. By default, aspect ratio is taken as per the Viewport size.

5. View Type: Sets the camera type to perspective or orthogonal.
6. OffsetX: Sets the lens offset values of the camera in X-axis. Works when FOV camera is OFF.
7. OffsetY: Sets the lens offset values of the camera in Y-axis. Works when FOV camera is OFF.
8. Camera FOV: Toggles camera FOV to work with the custom specified lens offset value.
9. FOV XY: Specifies the FOV of the camera in Horizontal(X) and Vertical(Y) axis.
10. Principal Axis XY: Specifies the offset of principal Axis of the lens.
11. Flag FOV: Specifies the FOV from the Addin based on Physical Camera Lens. It is derived from the Camera Lens calibrations. - 0 = FOV XY, 1 = FOV X, 2 = FOV Y.
12. Addin Aspect Ratio: Specifies the aspect ratio of physical camera derived from the V-Set addin. It is derived from the Camera Lens calibrations.
13. Override Aspect Ratio: When enabled, over-rides the Physical camera-based aspect ratio and allows user to manually specify the aspect ratio.
14. Conversion: It is used to convert principal axis from Millimeters to Pixel. It is used when the Principal Axis value is available in Millimeters from the V-Set Addin.
15. Data Table Row Index: Specifies the row in data table from which param of virtual camera are wired.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Omni



Create > Cameras and Lights > Omni



Click on the (Omni) icon in the Cameras and Lights tool group.

Standard Parameters

Standard Parameters		
>	Light Color	255,255,255,255
>	Attenuation	300.00,800.00
	Intensity	1.00
	Decay Type	Inverse Square

1. Light Color: Sets the color for the light.
2. Attenuation: Sets the Strength of Light
3. **Intensity:** Sets the Quantity of the Visible Light
4. Decay Type: Sets the Type of Decay to be applied..

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Particle System

This is an extension tool. It can be defined as a collection of objects, which are emitted from a source and their motion is controlled by certain parameters. It is used to visually simulate environmental elements like smoke, fire, bubbles etc. The particle system is based on point primitive (plane) emission, with texture map coordinates assigned on the particle primitive.



Create > Extensions > Particle System

Click on the



(Particle System tool) from the Extensions tool group.

Benefits of Particle System

Particle Systems can be used to create special effects like explosions, smoke, snow, fire, etc..

Parametric Attributes

Particle Parameters

Particle Parameters		
	Particle Size	1.00
	Size Variations	30
	Particle Type	POINTTYPE
	Particle Object	None
	Particle to Release	10
	Particle No. Variation	20
	Particle Release Interval	4
	Particle Life Cycle	60
	Maximum Particle	-1
	Emission	<input checked="" type="checkbox"/>
	Particle Data	<table trackkey="8"><dat...
	Reference	None
	Reference Type	NONE

1. Particle Size: Specifies the target size for all particles, depending on the type of particle.
2. Size Variations: Specifies the variation in size of the particles.

3. Particle Type: Specifies the Type of particle to be emitted.,
 - POINTTYPE: Specify the Type of particle to be emitted.
 - TRIANGLE: Emits the triangular shape particles.
 - QUAD: Emits the quad shape particles.
 - OBJECT: Allows selecting an object as particle.
4. **Particle Object:** Specify the object from the scene to be used as particle, works when object is selected as particle type.
5. Particle To Release: Specifies a fixed number of particles emitted per frame.
6. Particle No. Variation: Applies a variation value in the Number of Particles to release per frame.
7. Particle Release Interval: Specifies the intervals in particle generation in number of frames.
8. Particle Life Cycle: Sets the lifespan in number of frames of each particle from the frame of creation.
9. Maximum Particle: Specifies the maximum number of particles in the scene.
10. **Emission:** Enables or Disables the particle emission.
11. Particle Data: Wire DataSource.
12. Reference: Particle moves facing towards the selected reference object.
13. **Reference Type:** Specify the use of reference object

Emitter Parameters

Emitter Parameters		
	Emitter Type	POINTTYPE
	Emitter Object	None
	Number of Vertices	0
	Offset in Vertices	-1

1. Emitter Type: Choose to select Point or Object as particle emitter.
2. Emitter Object: Specify the object to emit the particles.
3. Number of Vertices: Specifies the area (in number of vertex) of object to be used as particle emitter. (-1 = All Vertex)
4. Offset in Vertices: Specifies the offset in the particle emission.

Emission Attributes

Enables or Disables the particle emission

Emission Attributes		
	Spread (degrees)	15.00
	Speed	250.00
	Speed Variation (%)	10
	Grow	20.00
	Grow Variation (%)	10
	Decay	20.00
	Decay Variation (%)	10
➤	Spin	0.00,0.00,0.00
	Spin Variation (%)	10
	Shrink (%)	0.00

1. Spread (Degrees): Sets the angular spread area of the particles.
2. Speed: Sets the velocity of the emitted particles.

3. Speed Variation (%): Sets a variation to the speed of emission for each particle.
4. Grow: Sets the percentage duration which the particle grows from being small to the particle Size value.
5. Grow Variation (%): Specifies a variation in growth time of each particle.
6. Decay: Sets the fade duration of particle during its lifetime.
7. Decay Variation (%): Sets a variation to the Decay of each particle.
8. Spin: Sets the velocity of spin of a particle in each direction(X,Y,Z). If set to 0, no rotation takes place.
9. Spin Variation (%): Sets the variation in the velocity of spin in each direction(X,Y,Z) for each particle.
10. Shrink: Sets the shrinkage of particle according to its life time.

Absorber Parameters

Absorber Parameters		
	Particle Absorber	None
	Absorber Sequence	<input type="checkbox"/>

1. Particle Absorber: Specifies an object from the scene which will ingest all particles.
2. Absorber Sequence : When checked on, the particles are pulled towards by every vertex of the absorber object as per their normalized sequence in the mesh. Else they are absorbed in a random order.

Particle Timing

Particle Timing		
	Start delay	0
	End Delay	0

1. Start Delay: Specifies the duration in percentage of particle life for which each particle will stay at emitter position.
2. End Delay: Specifies the duration in percentage of particle life for which each particle will settle on absorber before decaying.

Particle Dynamics

Particle Dynamics		
	External Forces	None
>	Gravity	0.00,-2.80,0.00
>	Wind	0.00,0.00,0.00
	Air Resistance	1

1. External Forces: Specifies the External Force objects to inherit the particle emission.
 - Gravity: Simulates the effect of gravity(in X,Y,Z direction) on emitted particles.
 - Wind: Simulates the effect of Wind blowing(in X,Y,Z direction) on emitted particles.
 - Air Resistance: Shows the friction of the particle. Default fixed to 1.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Blast

This is an extension tool for creating real-time blast effect with control over Gizmo movement and related dynamics.



Create > Extensions > Blast

Click on the (Blast tool) from the Extension tool group.

Benefits of Blast

- Creating Real-Time Blast Effect
- Control over Gizmo movement and related dynamics

Parametric Attributes

Explosion Parameters

Explosion Parameters		
	Source Object	None
➤	Bomb Location	0.00,0.00,0.00
	Explosion Radius	100.00
	Intensity	0.00
	Flake Size (Min)	1
	Flake Size (Max)	5
	Flake Extrude	1.00
➤	Spin	0.00,0.00,0.00
	Variation in Spin	0
	Group Node	Single Node

1. Source Object: Object to be exploded.
2. Bomb Location: Location of center of explosion in 3D space.
3. Explosion Radius: Effective distance of explosion.
4. Intensity: Sets the power of the bomb. Larger values make the particles fly farther.
5. Flake Size (Min): Specifies the minimum number of triangles per fragment.
6. Flake Size (Max): Specifies the maximum number of faces per fragment.

7. Flake Extrude: Sets the depth to the fragments.
8. Spin: Sets the intensity at which fragments rotate in each(X,Y,Z) direction.
9. Variation In Spin: Applies a variation to the Spin velocity of fragments.
10. Group Node: Choose to apply blast on the Parent node in a group node or the entire group node.

Explosion Dynamics

Explosion Dynamics	
>	Gravity 0.00,0.00,0.00
	External Forces None

1. Gravity: Simulates the effect of gravity(in X,Y,Z direction) on blast.
2. External Forces: Specifies the External Gravity or Wind force to be applied to the blast. Works in addition to the native gravity and wind.

Default

Default	
	Texture Modifier None
	Rotation Order XYZ
	Tag

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Tile

It is an extension tool to easily display a design in the tabular form rather than creating and arranging several 3D or 2D objects to achieve the same result. We typically create a table by assigning a Rectangle/Cube/3D Grid or any 3d or 2d object of different shape that respectively defines the design of each cell.

Uses of Tile Object: Match tally, batting card, league tables, candidate or team listing, lineups, weather plate etc.



Create > Extensions > Tile

Click on the  (Tile tool) from the Extensions tool group.

Benefits of Tile

Using Tile object helps us avoid creating several 3D/2D objects, which in turn keeps the render time under check.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Tiling Object	None
	Width	1,000.00
	Height	600.00
	Depth	1.00
	Tile Rows	10
	Tile Columns	3
➤	Tile Grout	0.00,0.00
	Number of Tiles	30
	Row Col Selection	<rowcol grout="0.000000,0...
	Display DataSet	<input type="checkbox"/>
	Global Texture	<input checked="" type="checkbox"/>

1. Tiling Object: Specifies the reference Object or Shape to be displayed as Tile.
2. Width: Sets the size along the X-axis.
3. Height: Sets the size along the Y-axis.
4. Depth: Sets the Depth if a 3D object is used a reference.
5. Tile Rows: Sets the number of Row divisions.
6. Tile Columns: Sets the number of column divisions.
7. Tile Grout: Sets the Gap between the tiles.
8. Number of Tiles: Display the Number of Tiles (Row x Columns).
9. Row Col Selection: Provides and interface to visually size the Row and Columns of the ble. user can opbonally Combine multiple Rows/Columns b using Fuse Row Column option.
10. Display DataSet: Property to make the Dataset visible in Variable sheet window.
11. Global Texture: Property to make he texture global/local to each tile/cell.

Tile Animation Effect

Tile Animation Effects		
	Animate Tile	1.0000,0.0000,0.0000,0.0000
	Texture	

1. Animate Tile: Animate each block of the tile individually using the interpolation of the SRTV key frames.
2. Texture: Specifies the texture to be used for Texture-Wise tile animation. The tiles animate according to the Luminance (darkest region to lightest) of the selected texture.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.

2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

E-Text

Use E-Text to create text blocks or text with special items like emoticons (animated or non-animated) for use in SMS or chat templates. You can create your emoticons as SWF (for animated) or PNG (for Non-Animated) and use them by assigning a special character combination. All the text related parameters like font, font size, kerning, or other text parameters are taken from the reference (selected) 3D Text.



Create > Extensions > E Text

Click on the  (E Text tool) from the Extensions tools group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Choose 3DText	None
	Block Mode	<input type="checkbox"/>

1. Choose3DText: Specify the reference 3D Text.
2. Block Mode: Toggles the E Text to be displayed as blocks or with text and emoticons.

Extended Parameters

Extended Parameters		
	Symbol Texture	None
	Rows	1
	Columns	1

1. Symbol Texture: Sets the Gris Texture for the E Text, Individual cell of this texture gets mapped to the E Text Symbol.
2. Rows: Specifies the number of Row of the emoticon texture grid.
3. Columns: Specifies the number of Columns of the emoticon texture grid.

Emoticon Options

Emoticon Options		
	Emoticon Mapping	<splsym><sym row="0" c
	Symbol Width	50.00
	Symbol Height	50.00

1. Emoticon Mapping: Assign individual expression for each cell of the grid to represent the desired emoticon in the selected texture.

2. Symbol Width: Specifies the width of the emoticon texture.
3. Symbol Height: Specifies the Height of the emoticon texture.

E-Text Animation Effects

E-Text Animation Effects		
	Animate	1.0000,0.0000,0.0000,0.0000
	Animation Texture	None

1. Emoticon Mapping: Assign individual expression for each cell of the grid to represent the desired emoticon in the selected texture.
2. Animation Texture: Specifies the texture to be used for Texture-Wise E-Text animate according to the Luminance (Darkest region to Lightest) of the selected texture.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. **Texture Modifier:** Specifies the Planer Texture Mapping object for Planer UVW texture mapping on the object.
2. **Rotation Order:** Allows user to specify the required Rotation order. It helps the user to assign the required order of Rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Extrude

This is an extension tool that can be used to create 3D extended mesh from a 2D shape input.



Create > Extensions > Extrude

Click on the  (Extrude Tool) from the Extensions tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Source Shape	None
	Extrude	50.00
	Segments (Height)	1
	Segments (Radial)	10

1. Source Shape: Allows selecting a source object by clicking Object button .
2. Extrude: Defines depth in the selected 2D source object.

- Segments (Radial): Sets the curve smoothness of the extruded object. The segments or steps of the 2D shape are not considered. Its value can be set from 1 to 40.
- Segments (Height): Sets the number of divisions along the extrude depth. Its value can be set from 1 to 100.

Extended Parameters

Extended Parameters		
> Skew	0.00,0.00	
Scale Base	0.00	
Twist Base	0.00	
Smoothing Angle	-1	
Inherit Source color	<input type="checkbox"/>	
Inherit Source Animation	<input type="checkbox"/>	
Fill	<input checked="" type="checkbox"/>	
Outline	<input type="checkbox"/>	

- Skew: Skew Object in X & Y Plane.
- Scale Base: Scale Base of Object.
- Twist Base: Twist Base of Object.
- Smoothing Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing.
- Inherit Source Color: When checked, Enabled Base Shape Color for Extrude.
- Inherit Source Animation: When checked, inherits the animation from he selected source object.
- Fill:** Toggles display of Fill,
- Outline: Toggles display of Outline,

Capping Options

Capping Options		
Top Cap	<input checked="" type="checkbox"/>	
Bottom Cap	<input checked="" type="checkbox"/>	
Start Cap	<input checked="" type="checkbox"/>	
End Cap	<input checked="" type="checkbox"/>	

- Top Cap: Toggles top cap visibility.
- Bottom Cap: Toggles bottom cap visibility.
- Start Cap:** Toggles display of start cap.
- End Cap:** Toggles display of end cap.

Texture Options

Texture Options		
Cap Texture	<input checked="" type="checkbox"/>	
Wall Texture	<input checked="" type="checkbox"/>	
> Cap UV Multiplier	1.00,1.00	
> Wall UV Multiplier	1.00,1.00	

- Cap Texture: Toggles the Top/Bottom Cap texture of the extruded object.
- Wall Texture: Toggles the wall texture of the extruded object.

3. Cap UV Multiplier: Specifies cap UV multiplier.
4. **Wall UV Multiplier:** Specifies wall UV multiplier.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Lathe

This is an extension tool that can be used to create a 3D mesh by revolving a 2D shape around a user-defined axis.



Create > Extensions > Lathe

Click on the  (Lathe tool) from the Extensions tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Source Shape	None
	Axis of Rotation	Y
➤	Centre of Rotation	0.00,0.00,0.00
	Segments (Radial)	15
	Segments (Height)	10
	Start Angle	0.00
	End Angle	360.00
	Angle	360.00

1. Source Shape: Specify the 2D Shape to apply Lathe.
2. Axis of Rotation: Sets the axis of rotation for the lathe.
3. Center of Rotation: Sets the center of the axis of rotation.
4. Segments (Radial): Sets the number of segment divisions or sides.
5. Segments (Height): Specifies the number of divisions between each edge of the Reference Shape. Overrides the actual steps value of the reference shape.
6. Start Angle: Sets the Angle around Y-axis from where the Slice starts.
7. End Angle: Sets the Angle around Y-axis from where the Slice ends.
8. Angle: Angle around Y-axis to define the angle extents.

Extended Parameters

Extended Parameters		
	Bounding Box Alignment	Centre
	Smoothing Angle	-1
	Inherit Source Animation	<input type="checkbox"/>

1. Bounding Box Center: Align the axis of revolution to the center or Minimum/Maximum extents of the shape.
2. Smoothing Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing.
3. **Inherit Source Animation:** When checked, inherits the animation from the selected source object.

Capping Options

Capping Options		
	Start Cap	<input checked="" type="checkbox"/>
	End Cap	<input checked="" type="checkbox"/>

1. Start Cap: Toggles display of Start Cap.
2. End Cap: Toggles display of End Cap

Texture Options

Texture Options		
	Mapping Mode	Start to End
>	UV Multiplier	1.00,1.00

1. Mapping Mode: Applies the texture scaling method to be applied to the material.
2. UV Multiplier: Specifies UV multiplier.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable placeholder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Bevel

This is an extension tool that can be used to create a 3D beveled mesh from a 2D shape. A bevel object extrudes a shape into 3D object and applies a flat or smooth bevel on the edges of the object.



Create > Extensions > Bevel

Click on the  (Bevel tool) from the Extensions tool group.

Benefits of Bevel

- Any 2D object can be converted into a 3D object.
- Bevel properties can be animated thus giving more options to your design.

- You can bevel any AI file. Bevel has more properties to give sharp edges to any 2D object which looks more effective and glossy on screen.
- With bevel properties, you can give a more realistic look to your object with intricate detailing.
- Beveled logo looks more effective during the animation.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Source Shape	None
	Bevel Height	5.00
	Bevel Inset	2.00
	Extrude Height	4.00
	Bevel Segments	3
	Power of Curve	1.00
	Segments (Source)	5
	Bevel CurveProfile	Power
	Inherit Source color	<input type="checkbox"/>
	Inherit Source Animation	<input type="checkbox"/>

1. Source Shape: Specify the 2D Shape to apply Bevel.
2. Bevel Height: Sets the Height of the beveled edge.
3. Bevel Inset: Sets the encroaching of the beveled wall into the shape.
4. Extrude Height: Sets the depth of the bevel shape toward the back.
5. Bevel Segments: Sets the number of segment divisions or sides of the bevel wall.
6. Power of Curve: Sets the intensity of the curve on the bevel wall.
7. Segments (Source): Specifies the number of divisions between each edge of the Reference Shape. Overrides the actual steps value of the reference shape.
8. Bevel Curve Profile: Choose the bevel curve for shape.
9. Inherit Source Color: When checked, Enabled Base Shape Color for Bevel.
10. **Inherit Source Animation:** When checked, inherits the animation from the selected source object.

Capping Options

Capping Options		
	Top Cap	<input checked="" type="checkbox"/>
	Bottom Cap	<input checked="" type="checkbox"/>
	Smooth Top Cap	<input type="checkbox"/>
	Smooth Bottom Cap	<input type="checkbox"/>
	Smooth Angle	-1
	Top Cap Material	None

1. Top Cap: Toggles display of top cap.
2. Bottom Cap: Toggles display of bottom cap.
3. Smooth Top Cap: When ON, applies a smoothing filter to the top cap polygons.
4. Smooth Bottom Cap: When ON, applies a smoothing filter to the bottom cap polygons.

5. Smooth Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing.
6. Top Cap Material: Specifies a separate material for the top cap. If empty, uses the default material applied to the object.

Texture Options

Texture Options		
	Top Cap Texture	<input checked="" type="checkbox"/>
	Bottom Cap Texture	<input checked="" type="checkbox"/>
	Wall Texture	<input checked="" type="checkbox"/>
>	Cap UV Multiplier	1.00,1.00
>	Wall UV Multiplier	1.00,1.00

1. Top Cap Texture: Toggles visibility of texture of the applied Material on Top face.
2. Bottom Cap Texture: Toggles visibility of texture of the applied Material on Bottom face.
3. Wall Texture: Toggles visibility of texture of the applied Material on bevel walls.
4. **Cap UV Multiplier:** Specifies cap UV multiplier.
5. **Wall UV Multiplier:** Specifies wall UV multiplier.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies weather the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Spline Extrude

This is an extension tool that can be used to create a 3D mesh from a 2D shape along a path. It can also inherit behavior from the cloth object. Spline Extrude creates a three-dimensional polygonal mesh by extruding a two-dimensional shape along an assigned path. Spline extrude is a Force Dynamics with no shape of its own and can be seen in Object Pool and Hierarchy window along with Standard material. A material is a design that you assign to the surface or face of an object so that it appears a certain way when rendered. Materials affect the color of objects, their glossiness, their opacity, and texture.. Spline Extrude inherits both shape and spline path using 2D primitive.



Create > Extensions > Spline Extrude



Click on the (Spline Extrude) from the Extensions tool group.

Benefits of Spline Extrude

- It can be used for dynamic designing in broadcast media.

Parametric Attributes

Standard Parameters

Standard Parameters	
Source Shape	None
Segments (Shape)	10
Source Spline	None
Segments (Spline)	10
Extrusion Axis	Y

1. Source Shape: Assigns a shape to the selected path or changes the current assigned shape.
2. Segments (Shape): Specifies the number of divisions between each edge of selected shape. Overrides the steps on the actual object.
3. Source Spline: Assigns a path to the selected shape or changes the current assigned path along which the Shape gets extruded.
4. Segments (Spline): Specifies the number of divisions between each edge of selected spline. Overrides the steps on the actual object.
5. Extrusion Axis: Orientation axis of the shape along the spline for extrusion.

Extended Parameters

Extended Parameters	
Start (%)	0.00
End (%)	100.00
Offset (%)	0.00
Smoothing Angle	-1
Inherit Source Animation	<input type="checkbox"/>

1. Start (%): Sets the start position of extrude along the spline.
2. End (%): Sets the end position of extrude along the spline.

3. Offset (%): Offsets the start and end position.
4. Smoothing Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing
5. **Inherit Source Animation:** When checked, inherits the animation from the selected source object.

Capping Options

Capping Options		
	Start Cap	<input checked="" type="checkbox"/>
	End Cap	<input checked="" type="checkbox"/>

1. Start Cap: Toggles display of Start Cap.
2. End Cap: Toggles display of End Cap

Texture Options

Texture Options		
	Mapping Mode	Start to End
>	UV Multiplier	1.00,1.00

1. Mapping Mode: Applies the texture scaling method to be applied to the material.
2. UV Multiplier: Specifies UV multiplier

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable placeholder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Table Text

Table Text is an extension tool to provide an easy way to display the associated categorical data in tabular form, reducing the hassle to create and arrange several 3D text objects to achieve the same result. We typically create a table text by assigning a tile object whose respectively defined dimensions of the rows and columns enable the table text to arrange itself in a tabular form on it.

The best use of Table Text: Match Tally, Batting Card, League Table, Candidate or Team Listing, Line-Up etc.



Create > Extensions > Table Text

Click on the  (Table Text) button in the Extensions tool group.

Benefits of Table Text

- It keeps the render time under check, since using table text helps us avoid creating several 3D text objects.
- Table text animation can be wired to its assigned tile animation and have character animations at the same time.
- The data can either be fed manually into each cell or can be wired to a third party data source with just a single drag and drop.
- Each column of data can have different font, size, color and type.
- It also has the option to merge two or more columns with different colors and font types using the 'Flexible' option.

Parametric Attributes

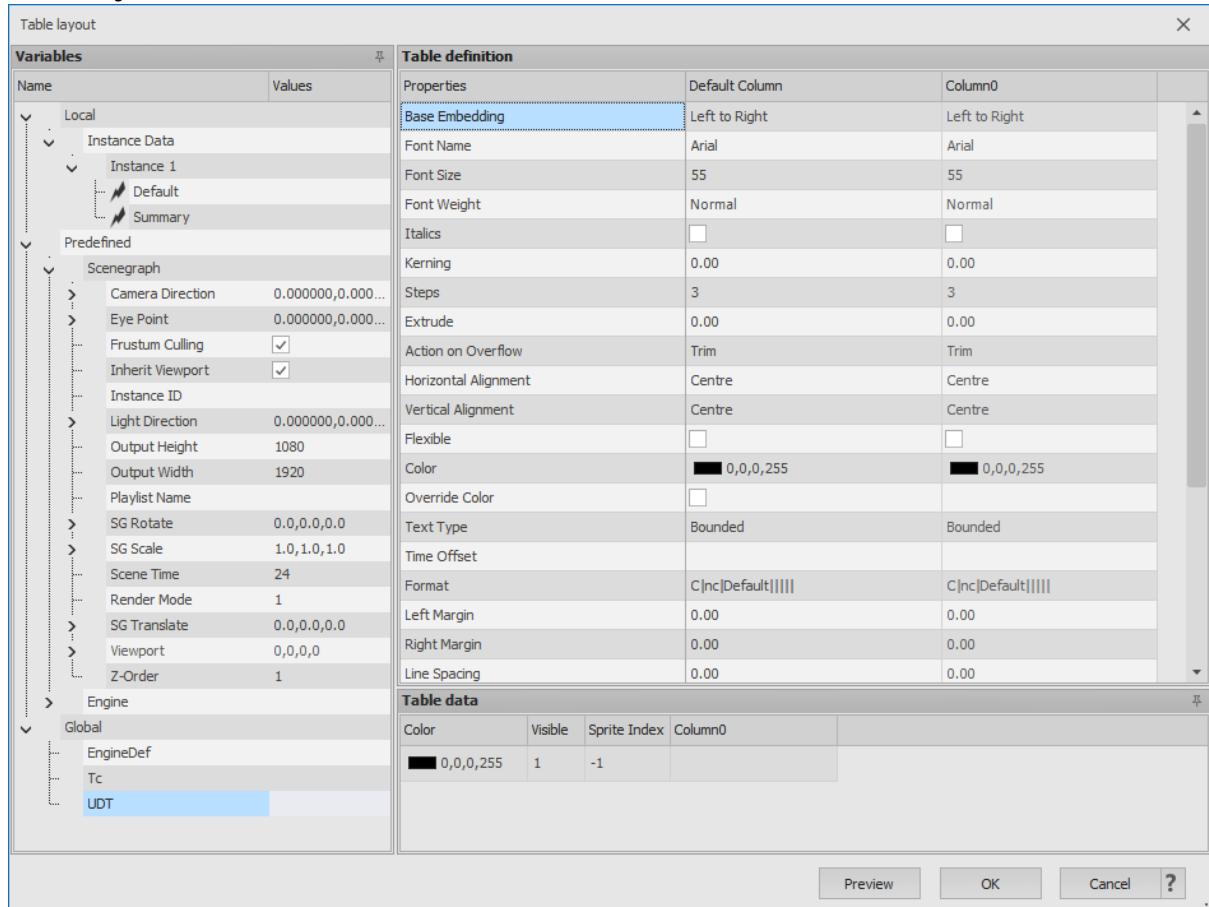
Standard Parameters

Standard Parameters		
	Source Object	None
	Table Data	<table><props><base ba... ...
	Wire Tile	<input type="checkbox"/>
	Detect Winding Order	<input type="checkbox"/>
	Tabletext Element Bound	100.00,60.00
	Width	100.00
	Height	60.00

1. Tile: Tile to be used for table layout.

2. Table Data: Data in each cell of Table.

Table Layout Window:



◦ Table Definition

- Base Embedding: Sets the Text embedding method. Choose Right to Left for Arabic.
- Font Name: Choose from a list of all available Windows fonts (.TTF and .OTF)
- Font Size: Sets the font size of the text. The size measurement method depends on the font.
- Font Weight: Sets the font Weight or boldness (Bold, Black, Thin etc.) of the text. Only weights supported by font will be displayed.
- Italic: Toggles italicized text.
- Kerning: Sets the distance between letters.
- Steps: Specifies the number of divisions between each edge of the Text.
- Extrude: Adds depth to the text.
- Action on Overflow: Sets the text adjustment method incase the amount of text exceeds the bounding area.
- Horizontal Alignment: Sets the horizontal orientation of the Text based on the cell.
- Vertical Alignment: Sets the vertical orientation of the Text based on the cell.
- Flexible: Concatenates two or more columns with different colors & font types.
- Color: Defines the diffuse color of the entire table text. Use Uniform color to see the effect as Standard Color will also get affected by Ambient and Specular values of the material.
- Override Color: Overrides the material applied on the object.

- **Text Type:** Choose from a list of Text3D formats like Integer, Digital Clock, Countdown Clock, Stopwatch etc.
- **Time Offset:** Sets the offset value to the Specified Time or Stopwatch type Text separated by colon e.g. h+2:m+30.
- **Format:** Specify format of the Clock/Date/Countdown/Stopwatch as per Windows Date and Time format method.
- **Left Margin:** Sets the offset from the left edge of the cell for the respective column.
- **Right Margin:** Sets the offset from the right edge of the cell for the respective column.
- **Line Spacing:** Adjusts the distance between lines of text. Effect only visible when text has multiple lines.
- **Align Axis:** Aligns the table text when Pie is used as tile object.
- **Sprite:** Displays a list of available textures from the texture pool to be used as sprite in table text or the respective column.
- **Sprite Width:** Defines the width of the selected sprite texture within the table cell.
- **Sprite Index:** Determines which sprite image from the texture is displayed in the selected column.
- **Fit Option:** Defines how the sprite image will be displayed within the cell. Fit ensures proportional scaling, Fill stretches to fill the cell, Image Size retains the original size, and another Fit option is available for proportional scaling.
- **Sprite Offset:** Adjusts the translation of the sprite image within the cell in the X, Y, or Z direction.

Text Effects

Text Effects		
	Character Effect	1.0000,0.0000,0.0000,0.0000

1. Character Effect : Character effect applied on table text

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.
2. **Tag:** Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Deflector Plane

This is one of the Force Dynamics tools, which enables creating a virtual surface to interact with Particle emission and provides realistic bounce back to the falling particles.

A single deflector plane can be assigned to multiple particle systems, and multiple deflectors can be assigned to a particle system to create a successive bounce. Deflector plane acts as a primary tool when creating real-time reflections on various surfaces.



Create > Force Dynamics > Deflector Plane

Click on the  (Deflector tool) from the Force Dynamics tool group.

Standard Parameters

Standard Parameters		
	Width	500.00
	Depth	500.00
	Dampness	20.00
►	Planar Reflection Params	100.00,5.00

1. **Width:** Sets the size along the X-axis
2. Depth: Sets the size along the Z-axis
3. **Dampness:** Controls the speed at which particles bounce off the deflector plane. At 0 particles bounce in same speed as they struck. At 100 they stick to the plane surface. Only works in Enterprise Version.
4. Planar Refiecton Params: Modifies the Planar Reflecton effect.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Gravity

This is one of the Force Dynamics tools, which enables creating gravity gizmo to affect extended objects such as Particle emission, Cloth, Blast etc.

Rotation of the Gravity gizmo in three axes detects the direction of the gravity effect. It can be used as an external force when creating special effects like Blast, Particle System etc.



Create > Force Dynamics > Gravity

Click on the  (Gravity Tool) from the Force Dynamics tool group.

Standard Parameters

Standard Parameters		
	Magnitude	5.00

Magnitude: Sets the intensity of the gravity force.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Wind

This is one of the Force Dynamics tools, which enables creating wind gizmo to affect extended objects such as Particle emission, Cloth, Blast etc.

Rotation of the wind gizmo in three axes detects the direction of the wind effect. It can be used as an external force when creating special effects like Blast, Particle System etc.



Create > Force Dynamics > Wind

Click on the  (Wind Tool) from the Force Dynamics tool group.

Standard Parameters

Standard Parameters		
	Magnitude	5.00

Magnitude: Sets the intensity of the wind force.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.

2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Group

The Group Node is a dummy object, which has its pivot at the center of the SceneGraph by default. As the name suggests, the Group Node is used to group or assemble multiple scene elements under a single node.

Adding objects to a Group simply requires selecting scene objects, which can then be dragged and dropped under the Group Node in the Hierarchy window. Objects added within a Group can be transformed and hence, animated independently without affecting the Group's hierarchy.

Nesting several Group Nodes is possible in the same way without affecting the user's control over children Group Nodes individually.



Create > Helper Objects > Create > Helper Objects > Group

Click on the **Group**  tool icon in the Helper Objects group.

EXAMPLE

Creating a Group Node and adding scene objects

1. Select the Sphere tool Sphere Tool and create a sphere.
2. Then select the Cube tool Cube Tool and create a cube.
3. Now click the Group button Group Node button
4. Press F9 from the keyboard to view the Hierarchy window. Here, you can see the objects created so far namely - Sphere 1, Cube 1, and Group 1.
5. Within the Hierarchy window, drag Sphere 1 and drop it under Group 1.
6. Following the same procedure, drag Cube 1 and drop it under Group 1.
7. Now you can see that these nodes have been added in Group 1.

Parameters

Default

Default	
Freeze Pivot	<input type="checkbox"/>
Opacity Node	None
Invert Opacity	<input type="checkbox"/>
Rotation Order	XYZ
Tag	

- Freeze Pivot: It locks the pivot of the group and does not allow it to change its position with addition or deletion of objects in the group.

- **Opacity Node:** Allows user to Define a node through which the objects in the group would be visible. The specified object serves as a window to see the group objects.
- Invert Opacity: Inverts the Node's Opacity
- Rotation Order: Allows user to specify the required Rotation Order. It helps user to assign the required order of Rotation for animating the object as per requirement.
- Tag: Serves as a variable place holder to hold values for the object. User can specify object names in Tag and wire the equivalent properties of the objects by specifying Object. Tag as Target.

Page Controllers

The Page Controller is a helper tool designed to facilitate the efficient display of diverse content across multiple pages. Paging technique allows users to showcase a variety of information re-using a collection of objects in the Page Controller. It presents and navigates through multiple pieces of content within a unified interface called Page Player. Users input data into the Data Paged control in the data entry form. The Page Controller processes the entered data, applying it to scene objects based on their variables and animates them. The Page Controller seamlessly animates through the data pages until all pages are displayed.

Users can add any number of pages, and the Page Controller dynamically integrates and animates elements with new data from each added page.

For example, if you need to show Top 10 news headlines, create a page control that can show 3 news items at one time. A scene will start its animation and pause at a specified time frame, however, page control starts its animation when set, executes its animation for number of pages in the control, here it is 4 and ends after reaching last page or repeat itself if directed.

Create > Helper Objects > Page Controllers

 Click on the  (Create Page Controller) tool icon button in the Helper Objects tool group.

Benefits of Page Controller

- Same number of design elements can be used to represent different sets of information.
- Simplified firing of back to back on-air graphics as pages can be played in automated mode with auto unload functionality.
- No programming or coding required for creation and playing of paged graphics.
- More elements can be added at a later stage in the paged group with a simple drag and drop, without affecting the existing elements.

Parametric Attributes

Default

Default	
Index	1
Start Time	0
Opacity Node	None
Invert Opacity	<input type="checkbox"/>
Tag	
Dataset	

1. **Index:** Specify the Index of the Page controller. It is best used when there are multiple page controllers in the scene. User can link the data Page controller of each Page Controller group using the Index number. It should be unique per scene.
2. Start Time: Specifies the Start frame of page controller.
3. **Opacity Node:** Allows user to Define a node through which the objects in the group would be visible. The specified object serves as a window to see the group objects.
4. **Invert Opacity:** Inverts the Node's Opacity
5. **Tag:** Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.
6. Dataset: This property sets the Page Control to acquire data pages from Addin when its data needs to be linked to an Addin..

Push Pin

Push Pin is a Helper object that allows sophisticated inter-linking of objects, giving users an advanced control over how dynamic elements behave in relation to each other. Thereby overcoming the limitations posed by simpler methods of Parent-Child hierarchical linking or Positional linking. Push Pin takes a Source Object as its primary input to establish interlinking between the scene elements.

All elements to be linked to the Source Object are required to be the Child nodes of the Push Pin. These Child nodes may maintain an intricate hierarchy of scene elements, groups etc. within themselves. It does not affect the functionality of the Push Pin in any way.



Create > Helper Objects > Push Pin

Click on the  (Push Pin tool) icon from the Helper Objects tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Source Object	None
	Distance	0.00,1.00,0.00
	X	0.00
	Y	1.00
	Z	0.00
	Pushpin Type	Linear
	Pushpin Scale	<input type="checkbox"/>
	Pushpin Rotate	<input type="checkbox"/>
	Pushpin Translate	<input type="checkbox"/>

- Source Object: Specifies the source object to which the Pushpin binds or links.
- Distance: Sets a normalized value of the distance from the Pivot of the Source object. At 0 pushpin will move to Pivot position. At 1 - to the extent of the object and at 0.5 at the mid of the object in the specified direction(X,Y,Z).
- Pushpin Type: Choose the application method of Pushpin to the source object. Linear - for position & rotation based, Angular - for angle based, Spherical - for geographical or Vertex- for specific vertex of an object or 2d shape.
- Pushpin Scale: When checked, the pushpin object applies the Scale values to the source object.
- **Pushpin Rotate:** When checked, the pushpin object applies the Rotation values to the source object.
- **Pushpin Translate:** When checked, the pushpin object applies the Translate values to the source object.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Scroller Group



Create > Helper Objects > Scroller Group

Click on the  (Scroller Group Node) Icon in the Helper Objects tool group.

Parametric Attributes

Default

Default		
	Opacity Node	None
	Invert Opacity	<input type="checkbox"/>
	Rotation Order	XYZ
	Tag	

1. **Opacity Node:** Allows user to Define a node through which the objects in the group would be visible. The specified object serves as a window to see the group objects.
2. Invert Opacity: Inverts the Node's Opacity
3. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
4. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Scroller

It is a tool which helps to convey up-to-date news information, most commonly seen on television, scrolling near the bottom of the screen.



Create > Helper Objects > Scroller

Click on the  (Scroller Node) Icon in the Helper Objects tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	> Velocity	-3.00,0.00,0.00
	Scroller Width	1,730.00
	Gap	30.00
	Marker 1	0.00
	Marker 2	100.00
	Initial Offset	0.00
	Pre Roll(frames)	0

1. Velocity: Specifies the speed of the Scroller in each direction (X,Y,Z).
2. Scroller Width: Specifies the size of scroll area.
3. Gap: Specifies the Gap between individual scroll groups. 0 means no space between the scroller group.

4. Marker 1: Declares the 1st Trigger position on the Scroller Node. User can use Tail and Head marker events in the On Scroller events to apply required actions for the Marker.
5. Marker 2: Declares the 2nd Trigger position on the Scroller Node. User can use Tail and Head marker events in the On Scroller events to apply required actions for the Marker.
6. Initial Offset: Specifies the starting position of the Scroller. When scroller is started the 1st scroller group would start from this location. 50 means the scroller group head position would be in center when the scroller is started.
7. **PreRoll (frames):** Sets a time(in frames) to trigger the pre-roll scroller event before the scroller item enters the scroll area.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Timer

It is a tool which helps to convey up-to-date news information, most commonly seen on television, scrolling near the bottom of the screen.



Create > Helper Objects > Timer

Click on the (Timer) Icon in the Helper Objects tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
Start/Stop	<input type="checkbox"/>	
Timer Type	One Shot	
Interval of Periodic Timer	0.00,0.00,0.00,1.00	
Hrs.	0.00	
Mins.	0.00	
Secs.	0.00	
Frames.	1.00	

1. Start/Stop: When Checked, starts the timer ticks.

2. Timer Type: Specifies the Timer type. One Shot will only trigger timer event ones, Periodic will trigger timer event every time it reaches the specified interval time. Alarm would trigger when the timer reaches the specified Alarm Data.
3. Interval of Periodic Timer: Specifies the list of calendar events to trigger Alarm type Timer.

UV Mapping

Planar Texture Mapping or UV Mapping is a procedure by which mapping is applied to a 3D object to define a relationship in the texture and geometrical surface. When UV mapping is done, it realigns the XYZ coordinates of the object in a new 3D space. The UV Co-ordinates can be divided as separate mesh in UVW space instead of XYZ space.



Create > Helper Objects > UV Mapping

Click on the  (Texture Mapping) Icon in the Helper Objects tool group.

Benefits of Planar Mapping

- It can be used to represent shadow mapping.
- It can be used to make transitions effectively.
- Can be used to make puzzle effect.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Type	Planar
	Width	200.00
	Height	150.00
	Smoothness	20.00
	Texture Scale	20.00

1. Type: Select the UV Mapping type. By default, it is Planar, which projects a texture along an axis onto a specified plane.
2. Width: Defines width of a planar on the X-axis. Default is 200. This width may start with minimum of 0 and go up to maximum of 9999.99.
3. Height: Defines height of a planar on the Y-axis. Default is 150. This height may start with minimum of 0 and go up to maximum of 9999.99.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.

2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Instancing



Create > Helper Objects > Create > Helper Objects > Instancing

Click on the **Instancing**  tool icon in the Helper Objects group.

Parameters

Objects

Standard Parameters	
	Target Shape <input type="text" value="None"/>
	Total Instances <input type="text" value="0"/>
	Instances per Contour <input type="text" value="1"/>
	Direction <input type="text" value="Increasing"/>
	Placement Mode <input type="text" value="PerContour"/>
	Element Animation <input type="text" value="1.0000,0.0000,0.0000,0.0000"/>
	Animation Texture <input type="text" value="None"/>
	Padded Animation <input checked="" type="checkbox"/>
	Overshoot <input type="checkbox"/>

1. **Target Shape:** Specify the 2D Shape to be set as instance target path.
2. **Total Instances:** Total instances to be rendered.
3. **Instances per Contour:** Instances to be rendered per contour
4. **Direction:** Placement direction of instances
5. **Placement Mode:** Specifies the Placement Mode of the Instances around target shape
6. **Element Animation:** Animate the instances individually using the interpolation of the SRTV key frames.
7. **Animation Texture:** Used with Texture-Wise character animation. The characters animate according to the Luminance (darkest region to lightest) of the selected texture.
8. **Padded Animation:** Toggle on/off padded animation.
9. Overshoot: Specify whether the instances should overshoot while animation.

Light Probe



Create > Helper Objects > Create > Helper Objects > Light Probe

Click on the **Light Probe**  tool icon in the Helper Objects group.

Parameters

Objects

Standard Parameters		
	Width	100.00
	Height	100.00
	Depth	100.00

1. Width: Sets the Edge of LightProbe
2. Height: Sets the Edge of LightProbe
3. **Depth:** Sets the Edge of LightProbe

Defalut

Default		
	Rotation Order	XYZ
	Tag	

1. Rotation Order: Allows user to speafy the required Rotation Order. It helps user to assign he required order of Rotation for animating the object as per requirement.
2. **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of å-te objects by specifying Object, Tag as Target.

3D Mesh

The Drone Designer application enables the users to import various types of objects including AI Shape, SHP Files, 3D Mesh, PSD File and FBX Nurbs.



Create > Import Objects > 3D Mesh

Click on the  (3D Mesh) from the Import Objects tool group.

File Attributes

File Attributes		
	File Name	C:\Users\rabi.narayan\Down

File Name: Show/Edit the file path of the 3DS file.

Extended Parameters

Extended Parameters		
	Smoothing Angle	30.00

Smoothing Angle: Smooths the faces based on the specified angle. -1 smooths all faces while 0 disable smoothing.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Ai Shapes

The Drone Designer application enables the users to import various types of objects including AI Shape, SHP Files, 3D Mesh, PSD File and FBX Nurbs.



Create > Import Objects > Ai Shape

Click on the  (Ai Shape) from the Import Objects tool group.

Standard Parameters

Standard Parameters		
	Bounded Width	200.00
	Bounded Height	200.00
	Segments	1
	Closed	<input type="checkbox"/>
	Stretch to Fit	<input type="checkbox"/>
	2-Sided	<input type="checkbox"/>
	Smoothing	0.00
	Display DataSet	<input type="checkbox"/>

1. **Bounded Width:** Specifies the Width of the AI when Stretch to Fit is Checked ON
2. **Bounded Height:** Specifies the Height of the AI when Stretch to Fit is Checked ON.
3. Segments: Specifies the number of divisions between each edge.
4. **Closed:** Joins the first and the last vertex with a straight line.
5. **Stretch to Fit:** Toggles AI to match extents to the Bounded Width and Bounded Height.
6. **2-Sided:** Toggles both side render of the 2D face or outline.
7. **Smoothing:** Applies Push & Pull effect to the Vertex handles.
8. Display Data Set: Property to make the Data set visible in Variable sheet window

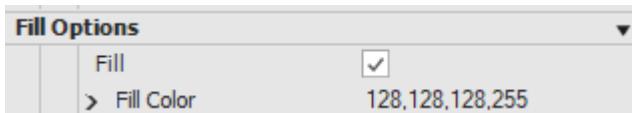
File Attributes

File Attributes		
	File Name	D:\Shape1.ai
	Polygon Data	

1. **File Name:** Show/Edit the file path of the AI file.

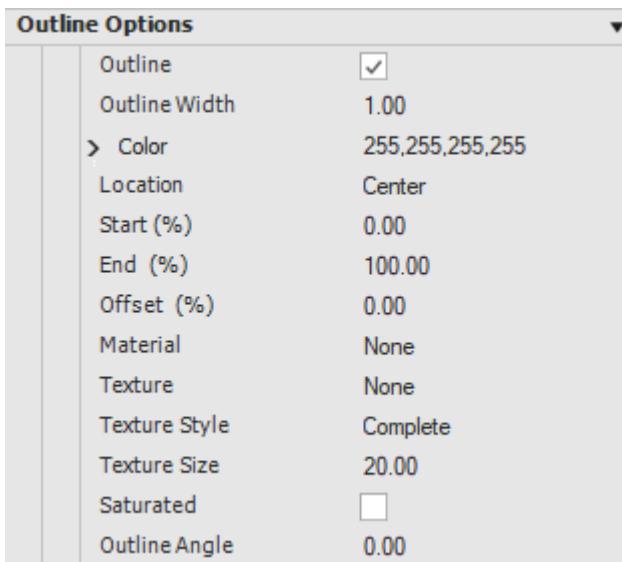
2. **Polygon Data:** Shows the list of named polygons in the AI File. User can hide/unhide or color individual Polys based on requirement. Suggested use - Parliamentary Seat Distribution or Map Coloring as per Party.

File Options



1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline Options



1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Starts (%): Sets the outline start position along the spline.
6. End (%): Sets the outline end position along the spline.
7. Offset (%): Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. Texture Size: Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. **Outline Angle:** Defines the rotation angle of Outline.

Advanced

Advanced	
	Use AI Vertex Bounding... <input type="checkbox"/>
	Vertex Shift
	AI Vertex Data
	Alpha Write <input checked="" type="checkbox"/>
	UVW Texture <input type="checkbox"/>
	Render Outline First <input type="checkbox"/>
	Regenerate Texture <input type="checkbox"/>
>	Tabletext Element Bo... 100.00,60.00

1. **Use AI Vertex Bounding Box:** Re-creates the bounding box of the AI shape as defined in the AI Vertex Data. Default = Unchecked
2. **Vertex Shift:** Use this to specify command to move the vertex position in runtime based on Adobe Illustrator shapedata command format.
3. AI Vertex Data: "User can plot data of an AI file using this property. Wagon Wheels shown in cricket are made using data in AI based format.
4. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
5. UVW Texture: Texture apply on complete shape.
6. **Render Outline First:** Render Outline Behind Object.
7. **Regenerate Texture:** If ON, texture will change along with the poly movement.
8. Tabletext Element Bounds: Specifies the Width!Height of the parent Tabletext elemnet cell

AI Animation

AI Animation	
	Character Effect 1.0000,0.0000,0.0000,0.00...
	Animation Texture None

1. **Character Effect:** Animate Poly of AI Shape individually using the interpolation of the SRTV key frames.
2. **Animation Texture:** Used with Texture-Wise character animation. The characters animate according to the Luminance (darkest region to lightest) of the selected texture.

Instancing Options

Instancing Options	
	Target None
	Range -1
	Wire Animation <input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies weather the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Specifies the Planar Texture Mapping object for planar UVW texture mapping on the objects.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

FBX Nurbs

The Drone Designer application enables the users to import various types of objects including AI Shape, SHP Files, 3D Mesh, PSD File and FBX Nurbs.



Create > Import Objects > FBX

Click on the  (FBX) from the Import Objects tool group.

File Attributes

File Attributes		
	File Name	C:\Users\rabi.narayan\Down

File Name: Show/Edit the file path of the FBX file.

Extended Parameters

Extended Parameters		
	Tessellation	6

Tessellation: Sets the amount of segment triangulation in the surface of the object.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

PSD

The Drone Designer application enables the users to import various types of objects including AI Shape, SHP Files, 3D Mesh, **PSD File** and FBX Nurbs.



Create > Import Objects > PSD File

Click on the (PSD File) from the Import Objects tool group.

File Attributes

File Attributes		
	File Name	C:\Users\rabi.narayan\Desktop\

1. File Name: Show/Edit the file path of the **PSD** file.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

SHP Files

The Drone Designer application enables the users to import various types of objects including AI Shape, SHP Files, 3D Mesh, PSD File and FBX Nurbs.



Create > Import Objects > SHP Shape

Click on the (SHP File) from the Import Objects tool group.

Standard Parameters

Standard Parameters		
	Bounded Width	500.00
	Bounded Height	500.00
	Segments	1
	Threshold	0.00
	Stretch to Fit	<input type="checkbox"/>
	2-Sided	<input type="checkbox"/>

1. Bounded Width: Specifies the Width of the SHP when Stretch to Fit is Checked ON.
2. Bounded Height: Specifies the Height of the SHP when Stretch to Fit is Checked ON.

3. **Segments:** Specifies the number of divisions between each edge.
4. **Threshold:** Sets the intensity of vertices on the path.
5. **Stretch to Fit:** Toggles SHP to match extents to the Bounded Width and Bounded Height.
6. **2-Sided:** Toggles both side render of the 2D face or outline.

File Attributes

File Attributes		
	File Name	C:\Users\rabi.narayan\Down...
	Polygon Data	<table/>
	Geo Referenced	<input type="checkbox"/>

1. File Name: Show/Edit the file path of the SHP file.
2. **Polygon Data:** Shows the list of named polygons in the AI File. User can hide/unhide or color individual Polys based on requirement. Suggested use - Parliamentary Seat Distribution or Map Coloring as per Party.
3. Geo Reference: Toggles the positioning of the SHP file according to the GIS coordinates.

Fill Option

Fill Options		
	Fill	<input checked="" type="checkbox"/>
	Fill Color	255,255,255,255

1. Fill: Hides or unhides the Fill area.
2. Fill Color: Sets the Fill color.

Outline Options

Outline Options		
	Outline	<input type="checkbox"/>
	Outline Width	2.00
	Color	255,0,0,255
	Location	Center
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>

1. Outline: Hides or unhides the outline.
2. Outline Width: Defines the thickness of the outline.
3. Color: Sets the outline color.
4. Location: Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. End (%): Sets the outline end position along the spline.
7. Offset (%): Offsets the outline start and end position.
8. Material: Select from the list of materials available in the Material Pool to be applied to the Outline.

9. Texture: Select the texture to be applied to the outline from the local drive.
10. Texture Style: Applies the texture scaling method to be applied to the outline texture or material.
11. Texture Size: Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.

Point Render Options

Point Render Options		
	Point	<input type="checkbox"/>
	Point Size	1.00

1. Point: Toggles the rendering of Outline as Point.
2. Point Size: Sets the size of Points of the outline.

Advance

Advanced		
	Alpha Write	<input checked="" type="checkbox"/>
	Vertex Shift	<input type="checkbox"/>
	UVW Texture	<input type="checkbox"/>
	Render Outline First	<input type="checkbox"/>
	Regenerate Texture	<input type="checkbox"/>

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Vertex Shift: Use this to specify command to move the vertex position in runtime based on Adobe Illustrator shapedata command format.
3. **UVW Texture:** Texture apply on complete shape.
4. Render Outline First: Render Outline Behind Object.
5. Regenerate Texture: If ON, texture will change along with the poly movement.

SHP Animation

SHP Animation		
	Character Effect	1.0000,0.0000,0.0000,0.0000
	Animation Texture	None

1. Character Effect: Animate Poly individually using the interpolation of the SRTV key frames.
2. Animation Texture: Used with Texture-Wise character animation. The characters animate according to the Luminance (darkest region to lightest) of the selected texture.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.

2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Menu Group

Menu Group is an interactive tool. It is similar to a normal group with an add on functionality of creating loop able objects with different data sets. It works as a normal group when in play mode but for Menu Group functionality, it needs to be in step mode with changes in few additional parameters.



Create > **Interactivity Tools** > Menu Group

Click on the  (Menu Group) button in the Interactivity Tools group.

Default

Default	
Anim Offset	0
Anim Start Frame	0
Anim Range	0
Menu Repeat	<input checked="" type="checkbox"/>
Opacity Node	None
Invert Opacity	<input type="checkbox"/>
Tag	
Dataset	

1. **Anim Offset:** The offset between the frames of menu items is Animation Offset. If it is not correct, the spacing between menu items would not be same after the 1st set is scrolled out.
2. Anim Start Frame: It is the value of 1st frame of the 1st menu item in the Menu Group.
3. Anim Range: The total number of frames of one Menu Item of the Menu group. For e.g. in the carousel, a group is animated from Right to Left in 100 frames, so the Animation Range would be 100
4. **Menu Repeat:** Specifies if the menu items should loop after the entire data set is displayed in the scene.
5. **Opacity Node:** Allows user to Define a node through which the objects in the group would be visible. The specified object serves as a window to see the group objects.
6. **Invert Opacity:** Inverts the Node's Opacity
7. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.
8. Dataset: This property sets the Menu Group to acquire data pages from Addin when its data needs to be linked to an Addin.

2D Text

This Primitive tool enables creating various types of 2D texts. It supports multiple formats like Date, Time, Decimal, etc.

Use 2D Text to create splines in shape of text or a special character available in the respective font. The number of vertices in the character or text varies according to the font. It can use the TrueType (.ttf) fonts installed on your system.



Create > Lines and Shapes > 2D Text



Click on the (2D Text Tool) from the Lines and Shapes tool group.

Parametric Attributes

Text Attributes

Text Attributes		
	Text	Press F11 to enter text
	Text Type	Free
	2-Sided	<input type="checkbox"/>
	Numeric text	0.00
	Format	C:\nc Default
	Text Length	7
	Inner Text	2D Text
	Vertical Text	<input type="checkbox"/>

1. **Text:** Edit/Modify text of the Text3D object in text editor. Allows multi line text, press Enter after each line to start new line.
2. **Text Type:** Choose from a list of 2d Text formats like Free, Time, Date, and Number..
 - **Free:** It allows you to create a simple text object without 'Word Wrap' functionality.
 - **Time:** Set this text type to create a digital clock object, which shows the system's time. After selecting this attribute, set the relevant parameter under Time/Date Options section.
 - **Date:** Select this text type to create a date object and show system's date. After choosing the text type as 'Date', set the correct format under Time/Date Options section.
 - **Numbers:** Use this text type to create an Integer or a Float numeric Text, depending on the usage.
 1. At default state, the numbers are defined as an Integer.
 2. When the numbers are changed by pressing Control key, the numbers change in a Decimal value.
3. **2-Sided:** Toggles both side render of the 2D face or outline.
4. **Numeric Text:** Sets the numerical value of 3d Text as an Integer or Decimal Type.
5. **Format:** Specify format of the Free/Clock/Date as per Windows Date and Time format method.
6. **Text Length:** Shows the number of Characters in the text including space.
7. **Inner Text:** Edit/modify text of the Text2D object in rich-text editor. Allows multi line text, press Enter after each line to start new line.

8. Vertical Text: Toggles vertical text.

Font Attribute

Font Attributes	
Font Name	Arial Black
Size	60
Weight	Black
Italics	<input type="checkbox"/>
Tracking	0
Fore Color	255,255,255,255

1. Font Name: Choose from a list of all available Windows fonts(.TTF and .OTF)
2. Size: Sets the font size of the text. The size measurement method depends on the font.
3. Weight: Sets the font Weight or boldness(Bold, Black, Thin etc.) of the text. Only weights supported by font will be displayed.
4. Italics: Toggles italicized text.
5. Tracking: Sets the distance between letters.
6. Fore Color: Applies the flat base color of the text. It sets the initial color of the RTF text. If no material is applied, this is the color in output.

Alignment Attribute

Alignment Attributes	
Horizontal Alignment	Centre
Vertical Alignment	Centre
Line Spacing	0.00
Base Embedding	Left to Right

1. Horizontal Alignment: Sets the horizontal orientation of the Text based on the Pivot or Bounding box of the text.
2. Vertical Alignment: Sets the vertical orientation of the Text based on the Pivot or Bounding box of the text.
3. Line Spacing: Adjusts the distance between lines of text. Effect only visible when text has multiple lines.
4. **Base Embedding:** Sets the Text embedding method. Choose Right to Left for Arabic.

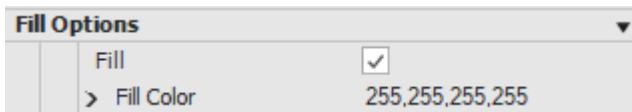
Bounding Text Options

Bounded Text Options	
Bounded Text	<input type="checkbox"/>
Show Bounded Rectangle	<input checked="" type="checkbox"/>
Action on Overflow	Fit
Width	100.00
Height	100.00

1. Bounding Text: Toggles text wrapping area constraints
2. Show Bounded Rectangle: Toggles display of Bounded Box(Purple Color) of the Bounded type Text.
3. Action On Overflow: Sets the text adjustment method in case the amount of text exceeds the bounding area.
4. **Width:** Specifies the Width of the Bounding Box of Bounded type Text

5. Height: Specifies the Height of the Bounding Box of Bounded type Text

Fill Options



1. Fill: Hides or unhides the Fill area
2. Fill Color: Sets the Fill color.

Outline Options

Outline Options	
Outline	<input type="checkbox"/>
Outline Width	1.00
Color	255,255,255,255
Location	Outer
Start (%)	0.00
End (%)	100.00
Offset (%)	0.00
Material	None
Texture	None
Texture Style	Complete
Texture Size	20.00
Saturated	<input type="checkbox"/>
Outline Angle	0.00

1. Outline: Hides or unhides the outline.
2. Outline Width: Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. Material: Select from the list of materials available in the Material Pool to be applied to the Outline.
9. Texture: Select the texture to be applied to the outline from the local drive.
10. Texture Style: Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. Saturated: Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of Outline.

Extended Parameters

Extended Parameters	
Steps	3
Alpha Write	<input checked="" type="checkbox"/>
UVW Texture	<input type="checkbox"/>
Render Outline First	<input type="checkbox"/>
Regenerate Texture	<input type="checkbox"/>
Detect Winding Order	<input type="checkbox"/>
Vertex Shift	<input type="checkbox"/>

- Steps:** Specifies the number of divisions between each edge of the Text.
- Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
- UVW Texture:** Texture apply on complete shape.
- Render Outline First:** Render Outline Behind Object.
- Regenerate Texture:** If ON, texture will change along with the poly movement.
- Detect Winding Order:** Auto detect the character(polygon) winding order
- Vertex Shift:** Use this to specify command to move the vertex position in run-time based on Adobe Illustrator shape data command format.

Instancing Options

Text Animation	
Character Effect	1.0000,0.0000,0.0000,0.00...
Animation Texture	None

Character Effect: Animate the characters of the text individually using the interpolation of the SRTV key frames.

Animation Texture: Used with Texture-Wise character animation. The characters animate according to the Luminance (darkest region to lightest) of the selected texture.

Instancing Options

Instancing Options	
Target	None
Range	-1
Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default	
Texture Modifier	None
Rotation Order	XYZ
Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.

2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Analog Clock

This 2D Primitive tool enables creating an analog clock.



Create > Lines and Shapes > Analog Clock

Click on the  (Analog Clock Tool) from the Lines and Shapes tool group.

Parametric Attributes

Standard Parameters

Standard Parameters	
Use System Clock	<input checked="" type="checkbox"/>
Time	16.00,34.00,20.00
Segments	1
2-Sided	<input type="checkbox"/>
Smoothing	0.00

1. **Use System Clock:** Toggles clock to get time from the local system.
2. Time: Specifies custom values for Hours, minutes and Seconds.
3. **Segments:** Specifies the number of divisions between each edge.
4. **2-Sided:** Toggles both side render of the 2D face or outline.
5. **Smoothing:** Smoothing to be applied on the edges of the analog clock.

Clock Attributes

Clock Attributes	
> Hour Hand	60.00,30.00,15.00,10.00
> Minute Hand	80.00,25.00,15.00,10.00
> Second Hand	100.00,5.00,20.00,10.00
Hour Hand Visibility	<input checked="" type="checkbox"/>
Minute Hand Visibility	<input checked="" type="checkbox"/>
Secondhand Visibility	<input checked="" type="checkbox"/>

1. **Hour Hand:** Sets the Shape Parameters for the Hour Hand
2. **Minute Hand:** Sets the Shape Parameters for the Minute Hand
3. Second Hand: Sets the Shape Parameters for the Seconds Hand
4. Hour Hand Visibility: Toggles visibility of Hour Hand.
5. Minute Hand Visibility: Toggles visibility of Minutes Hand.
6. **Secondhand Visibility:** Toggles visibility of Seconds Hand.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
	Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the fill area.
2. **Fill Color:** Sets the fill color.

Outline Options

Outline Options		
	Outline	<input type="checkbox"/>
	Outline Width	2.00
	Color	255,255,255,255
	Location	Center
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>
	Outline Angle	0.00

1. Outline: Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. Color: Sets the outline color.
4. Location: Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. Offset (%): Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. **Outline Angle:** Defines the rotation angle of Outline.

Advanced

Advanced		
	Alpha Write	<input checked="" type="checkbox"/>
	Render Outline First	<input type="checkbox"/>

Alpha Write: Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.

Render Outline First: Render Outline Behind Object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Callout



Create > Lines and Shapes > Callout

Click on the  (Callout Tool) from the Lines and Shapes tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Callout Text	None
	Align Anchor	Bottom Right
	Size Offset	10.00
	Rectangle Width	200.00
	Rectangle Height	150.00
	Chamfer Radius	10.00
	Segments	10
	2-Sided	<input type="checkbox"/>
	Smoothing	0.00

1. **Callout Text:** Specifies the reference of Text.
2. **Align Anchor:** Alignment of anchor.
3. Size Offset: Increase height and width of Callout object in proportion to the Text object.
4. Rectangle Width: Sets the size along the X-axis.
5. Rectangle Height: Sets the size along the Y-axis.
6. Chamfer Radius: Specifies the corner bevel radius.
7. **Segments:** Specifies the number of divisions between each edge.
8. **2-Sided:** Toggles both side render of the 2D face or outline.
9. **Smoothing:** Smoothing to be applied on the edges of the analog clock.

Pointer Option

Pointer Option		
	Pointer Offset	0.00
	Pointer Base	40.00
	Pointer Height	60.00

1. **Pointer Offset:** Sets the Offset of Anchor.
2. **Pointer Base:** Sets the base width of Anchor.
3. Pointer Height: Sets the height of Anchor.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
	Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline Options

Outline Options		
	Outline	<input type="checkbox"/>
	Outline Width	2.00
>	Color	255,255,255,255
	Location	Center
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>
	Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.

12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.

Advanced

Advanced		
	Alpha Write	<input checked="" type="checkbox"/>
	Vertex Shift	
	Render Outline First	<input type="checkbox"/>

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Vertex Shift: Use this to specify command to move the vertex position in runtime based on Adobe Illustrator shapedata command format.
3. Render Outline First: Render Outline Behind Object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

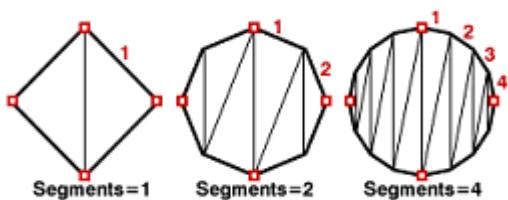
Circle

Circle is a 2D Primitive tool, which enables creating circular shape(s). A circle allows user to select its vertices and create different shapes.



Create > Lines and Shapes > Circle

Click on the  (Circle tool) from the Lines and Shapes tool group.



Parametric attributes

Standard Parameters

Standard Parameters		
	Radius	100.00
	2-Sided	<input type="checkbox"/>
	Smoothing	0.00
	Segments	20

1. **Radius:** Sets the radius.
2. **2-Sided:** Toggles both side render of the 2D face or outline.
3. **Smoothing:** Applies Push & Pull effect to the Vertex handles.
4. **Segments:** Specifies the number of divisions between each edge.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
	Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline Options

Outline Options		
	Outline	<input type="checkbox"/>
	Outline Width	2.00
	Color	255,255,255,255
	Location	Outer
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>
	Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.

9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of Outline.

Advanced

Advanced		
	Alpha Write	<input checked="" type="checkbox"/>
	Vertex Shift	<input type="checkbox"/>
	Render Outline First	<input type="checkbox"/>

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Render Outline First: Render Outline Behind Object.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. **Target:** Specify the instancing object to be used for instancing.
2. Range: Range of object instances.
3. Wire Animation: Specify whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

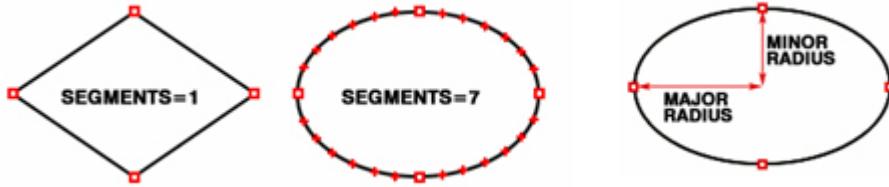
Ellipse

This 2D Primitive tool is a closed plane curve resulting from the intersection of a circular cone and a plane cutting completely through it.



Create > Lines and Shapes > Ellipse

Click on the  (Ellipse Tool) from the Lines and Shapes tool group.



Parametric attributes

Standard Parameters

Standard Parameters	
Radius (Minor)	100.00
Radius (Major)	150.00
Segments	20
2-Sided	<input type="checkbox"/>
Smoothing	0.00

1. Radius (Minor): Defines the radius of Ellipse along Y-axis.
2. Radius (Major): Defines the radius of Ellipse along X-axis.
3. Segments: Defines the number of segments in Ellipse.
4. 2-Sided: Renders the Ellipse from both sides when selected.
5. Smoothing: Smoothens the Ellipse and results change in shape.
6. Saturated: Limits the Outline Offset function. When selected, it defines the Outline Offset to be bound to 100% of the End Percentage. Unchecking the Saturated checkbox removes this limit.

Fill Options

Fill Options	
<input type="checkbox"/>	Fill
<input checked="" type="checkbox"/>	Fill Color 128,128,128,255

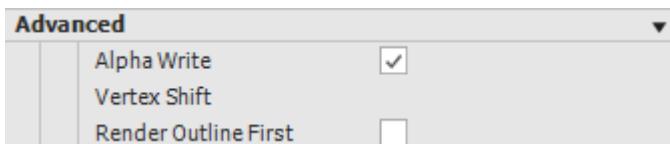
1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline



1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. End (%): Sets the outline end position along the spline.
7. Offset (%): Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of Outline.

Advanced



1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Vertex Shift: Use this to specify command to move the vertex position in run-time based on AI shape-data command format
3. Render Outline First: Render Outline Behind Object.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. **Target:** Specify the instancing object to be used for instancing.
2. Range: Range of object instances.
3. Wire Animation: Specify whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Freehand Polyline

This is a 2D Primitive tool. It allows creating freehand path with multiple vertices.



Create > Lines and Shapes > Freehand Polyline

Click on the  (**Freehand Polyline Tool**) from the Lines and Shapes tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Segments	12
	Closed	<input type="checkbox"/>
	2-Sided	<input checked="" type="checkbox"/>
	Smoothing	0.00

1. Segments: Specifies the number of divisions between each edge.
2. Closed: Joins the first and the last vertex with a straight line.
3. 2-Sided: Toggles both side render of the 2D face or outline.
4. Smoothing: Applies Push & Pull effect to the Vertex handles.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
	> Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline

Outline Options		
	Outline	<input type="checkbox"/>
	Outline Width	2.00
	> Color	255,255,255,255
	Location	Outer
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>
	Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. End (%): Sets the outline end position along the spline.
7. Offset (%): Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of Outline.

Advanced

Advanced		
	Alpha Write	<input checked="" type="checkbox"/>
	Vertex Shift	<input type="checkbox"/>
	Render Outline First	<input type="checkbox"/>

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Render Outline First: Render Outline Behind Object.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. **Target:** Specify the instancing object to be used for instancing.
2. Range: Range of object instances.
3. Wire Animation: Specify whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Gear

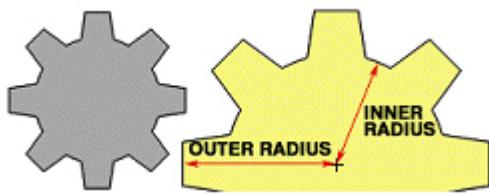
This is a 2D Primitive tool and allows creating toothed wheel.



Create > Lines and Shapes > Gear



Click on the (Gear Tool) from the Lines and Shapes Tool group.



Parametric attributes

Standard Parameters

Standard Parameters		
	Gears	6
	Radius (Inner)	100.00
	Radius (Outer)	140.00
	Hole	40.00
	Segments	5
	Gears Offset	0.00
	2-Sided	<input type="checkbox"/>
	Smoothing	0.00

1. Gears: Sets the number of cogs
2. **Radius (Inner):** Sets the distance of Inner edges from the center.
3. Radius (Outer): Sets the distance of outer edges from the center.
4. Hole: Sets the Hole radius.
5. Segments: Specifies the number of divisions between each edge.
6. Gears Offset: Rotates the inner set of vertices.
7. **2-Sided:** Toggles both side render of the 2D face or outline.
8. Smoothing: Applies Push & Pull effect to the Vertex handles.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
	Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline

Outline Options	
Outline	<input type="checkbox"/>
Outline Width	2.00
Color	255,255,255,255
Location	Outer
Start (%)	0.00
End (%)	100.00
Offset (%)	0.00
Material	None
Texture	None
Texture Style	Complete
Texture Size	20.00
Saturated	<input type="checkbox"/>
Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. **Start (%):** Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.

Point Render Options

Point Render Options	
Point	<input type="checkbox"/>
Point Size	1.00

1. Point: Toggles the rendering of Outline as Point.
2. **Point Size:** Sets the size of Points of the outline.

Advanced

Advanced	
Alpha Write	<input checked="" type="checkbox"/>
Vertex Shift	<input type="checkbox"/>
Render Outline First	<input type="checkbox"/>

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.

2. Vertex Shift: Use this command to move the vertex position in run-time based on Adobe Illustrator shape-data command format.
3. Render Outline First: Render Outline Behind Object.

Cloning Options

Cloning Options		
	Target	None
	Total Clones	0
	Clones per Contour	0
	Align	X
	Range	-1
	Overshoot	<input checked="" type="checkbox"/>
	Element Animation	1.000000,0.000000,0.000000...

1. **Target:** Specify the 2D Shape to be set as instance target path.
2. Total Clones: Total clones to be rendered.
3. Clones per Contour: Clones to be rendered per contour.
4. Align: Specifies the Alignment Axis of the Instances with target shape.
5. Range: Range of object instances.
6. **Overshoot:** Specify whether the instances should overshoot while animation.
7. Element Animation: Animate the clones individually using the interpolation of the SRTV key frames.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Helix

This is a 2D Primitive tool, and enables to create helix.



Create > Lines and Shapes > Helix

Click on the  (Helix Tool) from the Lines and Shapes Tool group.

Parametric attributes

Standard Parameters

Standard Parameters		
	Radius (Start)	0.10
	Radius (End)	200.00
	Height	200.00
	Turns	3
	Segments	10
	2-Sided	<input type="checkbox"/>
	Smoothing	0.00

1. Radius (Start): Sets the center offset of bottom edge.
2. **Radius (End):** Sets the center offset of top edge.
3. Height: Specifies the height.
4. Turns: Specifies the number of turns between the start and end.
5. Segments: Specifies the number of divisions between each edge.
6. 2-Sided: Toggles both side render of the 2D face or outline.
7. **Smoothing:** Applies Push & Pull effect to the Vertex handles.

Outline Options

Outline Options		
	Outline	<input checked="" type="checkbox"/>
	Outline Width	3.00
>	Color	255,255,255,255
	Location	Center
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>
	Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.

7. **Offset (%)**: Offsets the outline start and end position.
8. **Material**: Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture**: Select the texture to be applied to the outline from the local drive.
10. **Texture Style**: Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size**: Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated**: Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of outline.

Point Render Options

Point Render Options		
	Point	<input type="checkbox"/>
	Point Size	1.00

1. Point: Toggles the rendering of Outline as Point.
2. **Point Size**: Sets the size of Points of the outline.

Advanced

Advanced		
	Alpha Write	<input checked="" type="checkbox"/>
	Vertex Shift	

1. **Alpha Write**: Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Vertex Shift: Use this to specify command to move the vertex position in run-time based on Adobe Illustrator shape data command format.

Default

Default		
	Rotation Order	XYZ
	Tag	

1. **Rotation Order**: Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
2. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Line Graph

This is a 2D Primitive tool. It allows creating a real-time, data-fed line graph consisting of coordinates plotted on the X-axis and Y-axis (Plane). Each coordinate represents a curve vertex.



Create > Lines and Shapes > Line Graph

Click on the  (Line Graph Tool) from the Lines and Shapes Tool group.

Parametric attributes

Standard Parameters

Standard Parameters		
	Plot Graph	120 100 200 400 400 600...
	Graph Width	500.00
	Graph Height	300.00
	Segments	1
	Closed	<input checked="" type="checkbox"/>
	2-Sided	<input checked="" type="checkbox"/>
	Stepped Graph	<input type="checkbox"/>
	Gap	0.00
	Delimiter	

1. Plot Graph: Opens a dialog box to specify the graph data in X & Y coordinates, manually. It has a graphical user interface, to manipulate the graph values visually.
2. Graph Width: Specifies the Width.
3. Graph Height: Sets the size along the Y-axis
4. **Segments:** Specifies the number of divisions between each edge.
5. Closed: Joins the first and the last vertex with a straight line.
6. 2-Sided: Toggles both side render of the 2D face or outline.
7. **Stepped Graph:** Toggles the Line Graph to be displayed in Bar form or Line Graph form.
8. Gap: Sets the Gap between each bar in the Stepped Line Graph.
9. Delimiter: Specify the Line Graph Data separator in place of the | (pipe). If left empty, | would be used as the default separator.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
>	Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline

Outline Options	
Outline	<input type="checkbox"/>
Outline Width	2.00
Color	255,255,255,255
Location	Outer
Start (%)	0.00
End (%)	100.00
Offset (%)	0.00
Material	None
Texture	None
Texture Style	Complete
Texture Size	20.00
Saturated	<input type="checkbox"/>
Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. End (%): Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of Outline.

Point Render Options

Point Render Options	
Point	<input type="checkbox"/>
Point Size	1.00

1. Point: Toggles the rendering of Outline as Point.
2. **Point Size:** Sets the size of Points of the outline.

Advanced

Advanced	
	Alpha Write <input checked="" type="checkbox"/>
	Vertex Shift <input type="checkbox"/>
	Render Outline First <input type="checkbox"/>
	Use MinMax <input type="checkbox"/>
>	X Range 0.00,0.00
>	Y Range 0.00,0.00

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
 1. Vertex Shift: Use this to specify command to move the vertex position in run-time based on Adobe Illustrator shape-data command format.
 2. Render Outline First: Render Outline Behind Object.
 3. Use Min Max: Use Min Max properties.
 4. X Range: X range for graph.
 5. Y Range: Y range for graph.

Default

Default	
	Texture Modifier None
	Rotation Order XYZ
	Tag

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Line

This is a 2D Primitive Tool. It allows creating a path with two vertices.



Create > 2D Primitives > Line

Click on the  (Line Tool) from the Lines and Shapes Tool group.

Parametric attributes

Standard Parameters

Standard Parameters		
	Arrow	<arrow><edgepoint anchor="_
	Arrowhead	<input checked="" type="checkbox"/>
	Segments	12
	2-Sided	<input checked="" type="checkbox"/>
	Smoothing	0.00
	Arrowhead Multiplier(%)	100.00
	Arrow Length	100.00
	Angle	0.00

1. Arrow: Provides an interface to modify shape of the Arrowhead of the Line. Scroll Up and Down to zoom-in and out of the grid to make bigger arrows.
2. Arrowhead: Toggles visibility of Arrowhead on the end point.
3. Segments: Specifies the number of divisions between each edge.
4. 2-Sided: Toggles both side render of the 2D face or outline.
5. Smoothing: Applies Push & Pull effect to the Vertex handles.
6. Arrowhead Multiplier(%): Specifies the size of the Arrowhead.
7. Arrow Length: Length of Arrow.
8. Angle: Rotation angle around Z-Axis.

Fill Options

Fill Options		
	> Fill Color	255,178,0,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

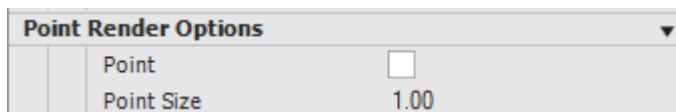
Outline

Outline Options		
	Outline	<input checked="" type="checkbox"/>
	Outline Width	3.00
	> Color	255,255,255,255
	Location	Center
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>
	Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.

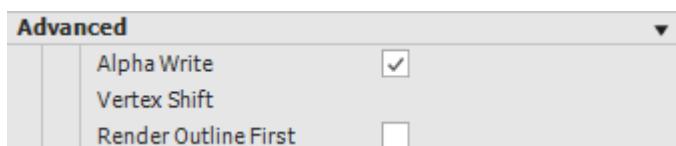
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of outline.

Point Render Options



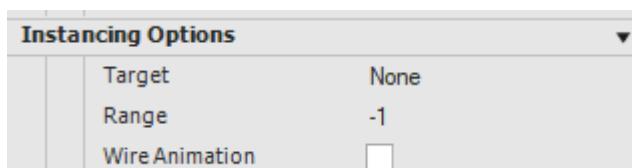
1. Point: Toggles the rendering of Outline as Point.
2. **Point Size:** Sets the size of Points of the outline.

Advanced



1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Vertex Shift: Use this to specify command to move the vertex position in run-time based on Adobe Illustrator shape-data command format.
3. Render Outline First: Render Outline Behind Object.

Instancing Options



1. **Target:** Specify the 2D Shape to be set as instance target path.
2. Range: Range of object instances.
3. Wire Animation: Specify whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Pie

This is a 2D Primitive tool, which enables creating Pie.



Create > Lines and Shapes > Pie



Click on the (Pie Tool) from the Lines and Shapes Tool group.

Parametric attributes

Standard Parameters

Standard Parameters		
	Object Type	Contiguous
	Inner Radius	100.00
	Outer Radius	150.00
	Pie Angle	360.00
	Pie StartPosition	0.00
	Slice Offset	0.00
	Pie Direction	CW
	Segments	6
	2-Sided	<input type="checkbox"/>
	Smoothing	0.00
	Pie Data	<table totalrows="5"><dat...

1. Object Type: Select the type of Pie from Contiguous, Discrete or Bar.
2. Inner Radius: Sets the Inner Radius of Pie.
3. Outer Radius: Sets the Outer Radius of Pie.
4. **Pie Angle:** Sets the total angle of the Pie object. Set 180 for semi-pie.
5. Pie Start Position: Specifies the angle at which the Pie would start formation.
6. Slice Offset: Offset between Pie Slices.

7. Pie Direction: Specifies the Clockwise or Counter-Clockwise orientation of the pie.
8. Segments: Specifies the number of divisions between each edge.
9. 2-Sided: Toggles both side render of the 2D face or outline.
10. Smoothing: Applies Push or Pull effect to the Vertex handles.
11. **Pie Data:** Allows user to provide data for the Pie object manually or through a data source or variable.

Animation Effects

Animation Effects		
	Element Animation	1.0000,0.0000,0.0000,0.00...
	Animate Angle	100.00
	Animation Align	Animate Start

1. Element Animation: Animate the Slices of the Pie individually using the interpolation of the SRTV key frames.
2. Animation Angle: Sets the total animation angle of the Pie object.
3. Animation Align: Sets the Start Position of the animation of the Pie Slices. Start- Animates from the Start Angle, Mid Animates from the center of Slices, End Animates from the End Angle.

Pin Parameters

Pin Parameters		
	Pin Position Angle and Depth	0.00,0.00,0.00
	Position Offset	0.00
	Angle Offset	0.00
	Depth Offset	0.00

Pin Position Angle and Depth: Specifies the location of the Pushpin object per slice of the pie.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
	Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline Options

Outline Options	
Outline	<input type="checkbox"/>
Outline Width	2.00
Color	255,255,255,255
Location	Outer
Start (%)	0.00
End (%)	100.00
Offset (%)	0.00
Material	None
Texture	None
Texture Style	Complete
Texture Size	20.00
Saturated	<input type="checkbox"/>
Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. **Start (%):** Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.

Point Render Options

Point Render Options	
Point	<input type="checkbox"/>
Point Size	1.00

1. Point: Toggles the rendering of Outline as Point.
2. **Point Size:** Sets the size of Points of the outline.

Advanced

Advanced	
Alpha Write	<input checked="" type="checkbox"/>
UVW Texture	<input type="checkbox"/>
Render Outline First	<input type="checkbox"/>
Regenerate Texture	<input type="checkbox"/>
Tabletext Element Bounds	25.00,25.00
Vertex Shift	

- Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
- UVW Texture: Texture apply on the complete shape.
- Render Outline First:** Toggles Render outline behind objects.
- Regenerate Texture: If ON, texture will change along with the Poly movement.
- Tabletext Element Bounds: Specifies the Width/Height of the Parent Tabletext element cell.
 - Height: Specifies the Height of the Parent Tabletext element cell.
 - **Width:** Specifies the Width of the Parent Tabletext element cell.
- Vertex Shift: Use this to specify command to move the vertex position in the run-time based on the Adobe Illustrator shape-data command format.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

- Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
- Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
- Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

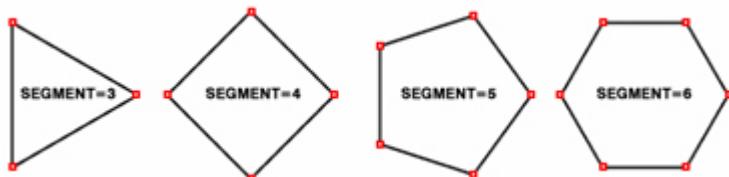
Polygon

This is a 2D Primitive tool, which enables creating polygon(s).



Create > Lines and Shapes > Polygon

Click on the  (Polygon Tool) from the Lines and Shapes Tool group.



Parametric Attributes

Standard Parameters

Standard Parameters		
	Sides	5
	Steps	2
	Radius	100.00
	2-Sided	<input type="checkbox"/>
	Smoothing	0.00

1. **Sides:** Sets the number of Sides.
2. **Steps:** Specifies the number of divisions between each edge.
3. **Radius:** Sets the Radius
4. **2-Sided:** Toggles both side render of the 2D face or outline.
5. **Smoothing:** Applies Push & Pull effect to the Vertex handles.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
	> Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline

Outline Options		
	Outline	<input type="checkbox"/>
	Outline Width	2.00
	> Color	255,255,255,255
	Location	Outer
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>
	Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.

8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of Outline.

Point Render Options

Point Render Options		
	Point	<input type="checkbox"/>
	Point Size	1.00

1. Point: Toggles the rendering of Outline as Point.
2. **Point Size:** Sets the size of Points of the outline.

Advanced

Advanced		
	Alpha Write	<input checked="" type="checkbox"/>
	Vertex Shift	<input type="checkbox"/>
	Render Outline First	<input type="checkbox"/>

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Vertex Shift: Use this to specify command to move the vertex position in runtime based on Adobe Illustrator shapedata command format.
3. Render Outline First: Render Outline Behind Object.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.

3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Polyline



Create > Lines and Shapes > Polyline

Click on the  (Polyline Tool) from the Lines and Shapes Tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Segments	12
	Closed	<input type="checkbox"/>
	2-Sided	<input checked="" type="checkbox"/>
	Smoothing	0.00

1. Segments: Defines the polyline segments.
2. Closed: Allows whether to join the start and end vertices of a polyline.
3. 2-Sided: Renders the polyline from both sides when selected.
4. Smoothing: Smoothens the polyline.
5. Saturated: Limits the Outline Offset function. When selected, it defines the Outline Offset to be bound to 100% of the End Percentage. Unchecking the Saturated checkbox removes this limit.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
	Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline Options

Outline Options	
Outline	<input type="checkbox"/>
Outline Width	2.00
Color	255,255,255,255
Location	Outer
Start (%)	0.00
End (%)	100.00
Offset (%)	0.00
Material	None
Texture	None
Texture Style	Complete
Texture Size	20.00
Saturated	<input type="checkbox"/>
Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. End (%): Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of Outline.

Point Render Options

Point Render Options	
Point	<input type="checkbox"/>
Point Size	1.00

1. Point: Toggles the rendering of Outline as Point.
2. **Point Size:** Sets the size of Points of the outline.

Advanced

Advanced	
Alpha Write	<input checked="" type="checkbox"/>
Vertex Shift	<input type="checkbox"/>
Render Outline First	<input type="checkbox"/>

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. **Vertex Shift:** Use this to specify command to move the vertex position in runtime based on Adobe Illustrator shapedata command format.
3. **Render Outline First:** Render Outline Behind Object.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Copy	<input type="checkbox"/>
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Copy: Specify whether the instances should be cloned or copied.
4. Wire Animation: Specifies weather the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Rectangle

This is a 2D Primitive tool, which enables creating Rectangle.



Create > Lines and Shapes > Rectangle

Click on the  (Rectangle tool) from the Lines and Shapes tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Rectangle Width	200.00
	Rectangle Height	150.00
>	Chamfer Radius	0.00,0.00,0.00,0.00
	Segments	1
	2-Sided	<input type="checkbox"/>
	Smoothing	0.00
	Base Pivot	Centre

1. Rectangle Width: Sets the size along the X-axis
2. Rectangle Height: Sets the size along the Y-axis
3. Chamfer Radius: Sets the chamfer radius.
4. Segments: Specifies the number of divisions between each edge.
5. 2-Sided: Toggles both side render of the 2D face or outline.
6. Smoothing: Applies Push & Pull effect to the Vertex handles.
7. Base Pivot: Shifts the origin of the Rectangle object. The Height parameter works according to Alignment.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
>	Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline Options

Outline Options		
	Outline	<input type="checkbox"/>
	Outline Width	2.00
>	Color	255,255,255,255
	Location	Outer
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>
	Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.

4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of Outline.

Point Render Options

Point Render Options		
	Point	<input type="checkbox"/>
	Point Size	1.00

1. Point: Toggles the rendering of Outline as Point.
2. **Point Size:** Sets the size of Points of the outline.

Advanced

Advanced		
	Alpha Write	<input checked="" type="checkbox"/>
	Vertex Shift	<input type="checkbox"/>
	Render Outline First	<input type="checkbox"/>

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Vertex Shift: Use this to specify command to move the vertex position in runtime based on Adobe Illustrator shapedata command format.
3. Render Outline First: Render Outline Behind Object.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It help user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable place holder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object. Tag as Target.

Star

This is a 2D Primitive tool, which enables creating Star.



Create > Lines and Shapes > Star

Click on the  (Star) tool from the Lines and Shapes tool group.

Parametric Attributes

Standard Parameters

Standard Parameters		
	Star Points	5
	Radius (Inner)	50.00
	Radius (Outer)	100.00
	Segments	5
	Star Offset	0.00
	2-Sided	<input type="checkbox"/>
	Smoothing	0.00

1. Star Points: Sets the number of Points.
2. **Radius (Inner):** Sets the distance of Inner edges from the center.
3. **Radius (Outer):** Sets the distance of outer edges from the center.
4. Segments: Specifies the number of divisions between each edge.
5. **Star Offset:** Rotates the inner set of vertices.
6. **2-Sided:** Toggles both side render of the 2D face or outline.
7. Smoothing: Applies Push & Pull effect to the Vertex handles.

Fill Options

Fill Options		
	Fill	<input checked="" type="checkbox"/>
>	Fill Color	128,128,128,255

1. **Fill:** Hides or unhides the Fill area.
2. **Fill Color:** Sets the Fill color.

Outline Options

Outline Options		
	Outline	<input type="checkbox"/>
	Outline Width	2.00
>	Color	255,255,255,255
	Location	Outer
	Start (%)	0.00
	End (%)	100.00
	Offset (%)	0.00
	Material	None
	Texture	None
	Texture Style	Complete
	Texture Size	20.00
	Saturated	<input type="checkbox"/>
	Outline Angle	0.00

1. **Outline:** Hides or unhides the outline.
2. **Outline Width:** Defines the thickness of the outline.
3. **Color:** Sets the outline color.
4. **Location:** Sets the outline position to center, Inner or Outer side of the fill area.
5. Start (%): Sets the outline start position along the spline.
6. **End (%):** Sets the outline end position along the spline.
7. **Offset (%):** Offsets the outline start and end position.
8. **Material:** Select from the list of materials available in the Material Pool to be applied to the Outline.
9. **Texture:** Select the texture to be applied to the outline from the local drive.
10. **Texture Style:** Applies the texture scaling method to be applied to the outline texture or material.
11. **Texture Size:** Specifies the size of texture in pixels to define tiling of texture along the outline when Tiled Texture Style is applied.
12. **Saturated:** Toggles outline offset limit. When the Outline-Offset limits to Start and End Points.
13. Outline Angle: Defines the rotation angle of Outline.

Point Render Options

Point Render Options		
	Point	<input type="checkbox"/>
	Point Size	1.00

1. Point: Toggles the rendering of Outline as Point.
2. **Point Size:** Sets the size of Points of the outline.

Advanced

Advanced		
	Alpha Write	<input checked="" type="checkbox"/>
	Vertex Shift	
	Render Outline First	<input type="checkbox"/>

1. **Alpha Write:** Toggles the rendering of Alpha Channel if no material is applied. Default = Checked.
2. Vertex Shift: Use this to specify command to move the vertex position in run-time based on Adobe Illustrator shape-data command format.
3. Render Outline First: Render Outline Behind Object.

Instancing Options

Instancing Options		
	Target	None
	Range	-1
	Wire Animation	<input type="checkbox"/>

1. Target: Specify the 2D Shape to be set as instance target path.
2. **Range:** Range of object instances.
3. Wire Animation: Specifies whether the instances use animation of this object or instancing object.

Default

Default		
	Texture Modifier	None
	Rotation Order	XYZ
	Tag	

1. Texture Modifier: Selects the Planar Texture Mapping object for texture mapping on the object.
2. **Rotation Order:** Allow user to specify the required Rotation Order. It helps user to assign the required order of rotation for animating the object as per requirement.
3. Tag: Serves as a variable placeholder to hold values for the object. User can specify the object name in TAG and WIRE the equivalent properties of the objects by specifying object Tag as Target.

Camera View



Viewport Tools > Camera View

Click on the (View) button.

This is an Expand button that allows users to switch between different viewport views, such as Front, Back, Top, Bottom, Left, and Right.

By holding down the button, a pane is displayed to select other views.

Full Screen



Viewport Tools > Full Screen

Click on the  (Full Screen Tool) button.

This toggle button expands the viewport in the Designer to its maximum size, hiding all other docks, windows, and toolbars for a distraction-free workspace.

Gestures



Viewport Tools > Gestures

Click on the  (Gestures) button.

When enabled, this mode allows users to perform various touch actions in the viewport, simulating how the scene would behave in a touch-enabled environment.

This is useful for testing interactive touch scenes. Right-clicking the tool reveals a list of other modes for testing various scenarios.

Grid



Viewport Tools > Grid

Click on the  (Grid Tool) button.

Toggles the visibility of a Grid overlay in the viewport, assisting users in object alignment and design accuracy by snapping objects to grid points.

Guides



Viewport Tools > Guide

Click on the  (Guide Tool) button.

This toggle button lets users place guides in the view port. Guides can only be added in the Camera View, not in orthogonal views. Vertical and horizontal guides can be positioned by clicking the inner left

or inner top of the view port. To assist with guide placement, users can enable the Ruler for easy identification of the click area.

Pan



Viewport Tools > Pan



Click on the (Pan Tool) button.

Activates the Pan Tool for orthogonal views, allowing users to move around the view by dragging. This tool does not work in Camera Views. Users can also press and hold the "P" key in orthogonal views to pan, even when the Pan button is not activated.

Quick Render



Viewport Tools > Quick Render



Click on the (Quick Render) button.

This button lets users Quickly Render a snapshot of the current frame of the scene. It supports image formats like BMP, PNG, JPEG, and TIFF.

Simply click to capture the current viewport as an image file.

Render to Disk



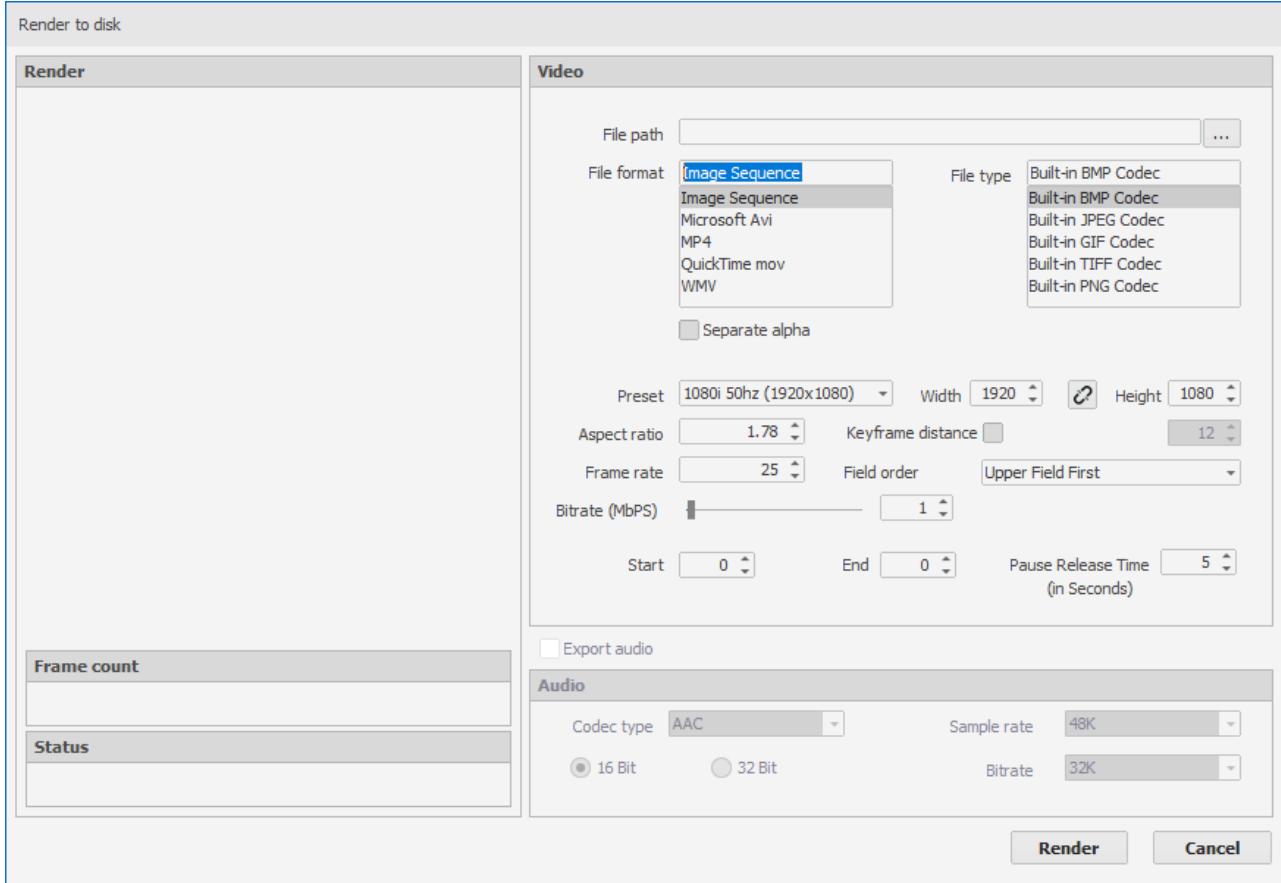
Viewport Tools > Render to Disk



Click on the (Render to Disk) button.

Opens the Render to Disk dialog box, where users can render the current scene as an image or video output.

This allows users to save their work in various formats for external use.



Ruler



Viewport Tools > Ruler

Click on the  (Ruler Tool) button.

Toggles the visibility of the Ruler along the edges of the viewport, helping users align objects and place guides with precision.

Zoom



Viewport Tools > Zoom

Click on the  (Zoom Tool) button.

Enables the Drag to Zoom Tool for orthogonal views. This tool allows users to zoom in and out by dragging the mouse. It does not work in Camera Views. Users can press and hold the "Z" key to zoom in orthogonal views with a mouse drag, even if the Zoom button is not selected.

Zoom Extent



Viewport Tools > Zoom Extent



Click on the (Zoom Extent Tool) button.

The Zoom Extent button adjusts the viewport to fit all the content in the scene. It automatically zooms in or out in orthogonal views to ensure that the entire scene content is visible within the viewport bounds.

Zoom Window



Viewport Tools > Zoom Window



Click on the (Zoom Window Tool) button.

This tool allows users to create a rectangular area in the orthogonal viewport. The selected area is zoomed in to fit the viewport, providing a focused view of that section.

SRTV Panel

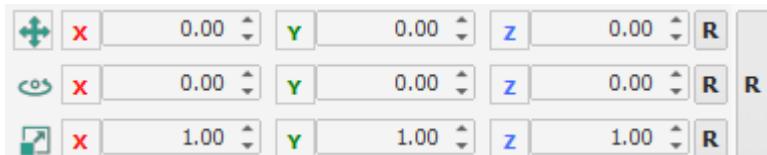
SRTV is an acronym for Scale, Rotation, Translation, and Visibility. It provides a quick access to the tools for design and setting of an object.



All the tools have been described in detail below:

SRT Panel

The SRT panel is used for transformation of an object in the form of scaling, rotation and transition within the three given axes - X, Y and Z.



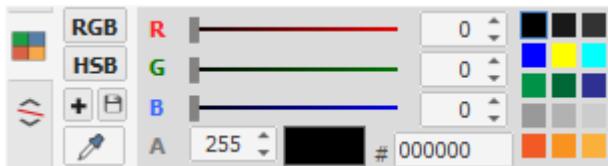
TRANSFORM Panel - List down the options do transformations on the objects created on viewport

- Translate: Change position/axis of the object.
- Rotate: Change orientation of the object.

- Scale: Change the size (x, y or z coordinates) of the object.
- Transformations can be performed on the newly created or selected object in a scene. The reference point for these transformations is the objects pivot-point

Color Palette

Color Palette is the finite set of colors provided within an application for management of digital images.



It allows the users to load and save a new color file (.clt) depending on the project requirements.

To use color from the palette, drag the color swatch and drop onto a material A material is a design that you assign to the surface or face of an object so that it appears a certain way when rendered. Materials affect the color of objects, their glossiness, their opacity, and texture. parameter (i.e. Amb, Diff, Spec, etc.)

It also allows the user to chose a color using the color picker.

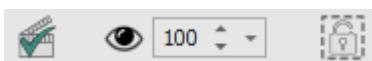
Modes



Sets the mode of the selected object to Object (o), Manipulator(m), Pivot(i), Vertices(v) mode in the viewport area. Drone Designer allows user to modify objects, their properties and parameters right from the viewport. Various modes provide varied functions to do the required changes to the selected object(s).

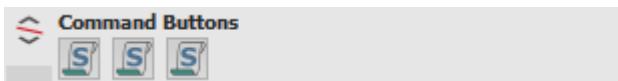
- **Object (O):** This mode is the default mode of Drone Designer. The object transformations can be done in the viewport in this mode.
- **Manipulator (M):** This mode provides additional helper gizmos to do parametric changes to the objects right from the viewport.
- **Texture Manipulator (T):** This mode allows user to translate and scale the texture on the selected object. This mode is activated by pressing the 'T' shortcut key when working in the Viewport.
- **Pivot (I):** This mode allows user to translate the pivot in the viewport. Right-Click on the pivot mode icon to display the pivot transformation dialog box.
- **Vertices (V):** This mode is show only for 2D objects. In this mode enables user to transform vertices of the selected 2D object. All changes done in the vertices mode are animate-able.
- **Polygon Mode (G):** Polygon mode allows you to select and transform individual polygons of a 2D Object. This mode is only available for objects that contain multiple polygons, such as 2D text, shapes imported from Illustrator, or shapes that have been combined using the Composite Shape option in Designer. If you want to apply a different fill color to a specific polygon, you can press the ALT key while dragging a color from the color swatch onto the polygon. This will apply the color to the selected polygon only, without affecting the other polygons in the object.

- **Program:** A tool used to show or hide the live programs to be displayed while the show is on air.
- Layout: User can select a scene to be used as a Layout. The Layout scene is rendered above scene and can be used as a reference to align or adjust the working scene. Users can right-click on a template (Local Scene or Template) and select Set LayoutSG option. When the scene is applied, the Icon changes to . Right-Click on the Layout Icon and select Clear Layout to remove the Layout SG.
- **Show Statistics:** (To display/hide set of information, press "ALT + C" or "ALT + C + C" from the keyboard) It portrays various render statistics information at bottom of the Scene Graph, which helps users to view all render statistics information collectively in the Statistics bar.
- Align: Used to align multiple selected objects so they line up on a specified axis.



- **Render** - Is a toggle button to Render On and Render Off selected object(s). When an object is rendered on, it shows a tick mark in green and when it is rendered off it shows a icon.
- **Visibility** - A tool used to show or hide an object.
- **Lock Selection** - A tool used to lock or unlock the position of an object.
- Play Control - A panel used to view the behavior of objects collectively when the show is on-air.

Custom Commands



The SRTV panel displays all Custom Commands written by the user and saved as Buttons for quick recall.

1. Objects Count: Displays the number of objects in the current scene.
2. Textures Count: Displays the number of textures in the current scene.
3. Materials Count: Displays the number of materials in the current scene.

Parameters

The Parameters Pool is a core utility within the editor interface, accessible via the F4 shortcut key. It dynamically updates based on the type and number of selected objects, displaying only common editable parameters across the selection. Each section of the Parameters Pool offers purpose-specific properties and controls to enhance object manipulation, animation, and real-time interaction.

The Parameters Pool is divided into five functional sections:

1. Object Section
2. Pivot Presets Tab
3. Node Tab

4. Display Properties Tab
5. Waspi Tab

Objects

Object Section

The Object section displays basic editable properties for the selected object(s). It supports multi-selection editing for objects of the same class, providing a unified control experience.

Parameter Linking Across Same Class Objects: If multiple objects of the same type are selected, the same parameter fields are shown for batch updates.

Right-Click Context Options on Parameters: Each parameter offers a right-click menu with the following options:

- Add to Quick Access: Adds the selected parameter to any of the three Quick Access slots shown in the Hierarchy. This lets users quickly change frequently used values.
- Favorites: For animated parameters, this saves their animation into the Favorites pool. It can be reused on other similar parameter types using drag-and-drop.
- Add User Tag: Available when the TrackView is at a keyframe. It adds a dynamic tag to the parameter, which can be bound to input fields in Data Entry Forms.
- Interpolator: Opens the list of available interpolators. Each interpolator modifies how values change between keyframes:
 - Bezier: Creates smooth, controllable animation curves with handle control.
 - Left Handle:
 - X, Y, Z: Sets position of the curve's start direction.
 - W: Adjusts the ease-in curve.
 - Right Handle:
 - X, Y, Z: Controls the end direction of the curve.
 - W: Adjusts the ease-out curve.
 - Boolean: Available for true/false parameters. Used for toggling visibility, switches, or activation flags.
 - Linear: Provides constant speed transitions between keyframes.
 - Noise: Introduces procedural randomness with defined constraints. Parameters include:
 - Noise Interval: Duration or spacing of the noise wave pattern.
 - Octaves: The number of noise layers added to refine complexity.
 - Amplitude: Strength or height of noise displacement.
 - Smoothness: Controls how fluid or jittery the noise appears.
 - Minimum X/Y/Z/W: Defines the lower bound of randomness per axis.
 - Maximum X/Y/Z/W: Defines the upper bound of randomness per axis.
 - Numeric: Used for numeric Text parameters where value progression is animated.
 - Path: Enables animation along a predefined curve or trajectory.
 - State: Switches between discrete states (e.g., Active/Inactive, On/Off).

- Wire: Binds this parameter to another parameter or a variable of matching data type (e.g., float3 to float3).

Animation Keyframe Indicators : Animated parameters are marked with a Key icon:

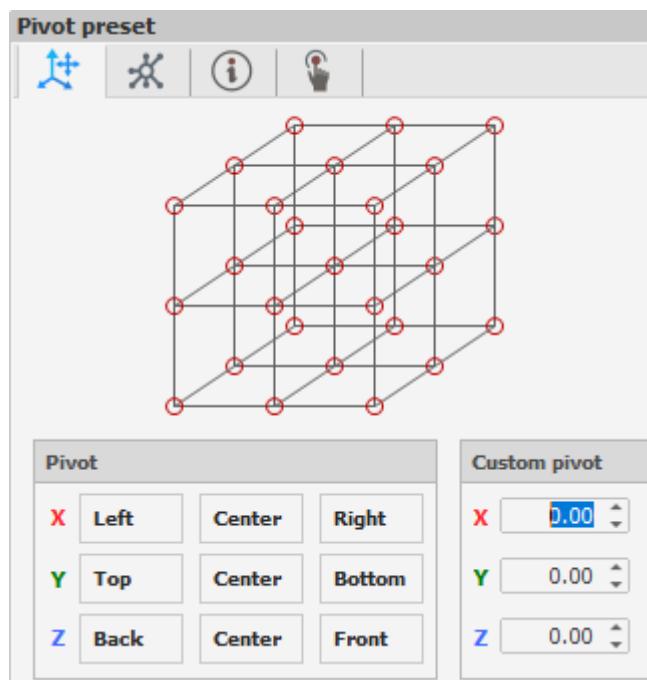
- Orange: Active keyframe at current timeline frame
- Grey: Keyframe exists but not on current frame

Additional animation-related right-click options:

- Remove All Keys: Deletes all keyframes for the parameter.
- Remove Key: Deletes only the keyframe on the current frame.
- Go To Previous Key: Moves the TrackView head to the previous keyframe.
- Go To Next Key: Moves to the next available keyframe.

Pivot Preset

The Pivot Preset Tab is designed to help users easily set the pivot point of an object to predefined or custom positions. Pivot placement affects how objects scale, rotate, and position in space.



Pivot Grid (27-Point Cube): At the center of the tab is a 27-point cube representing different pivot alignment options:

- X-Axis (Left to Right):
 - Left: Sets the pivot to the far left of the object's bounding box.
 - Centre: Aligns the pivot to the center along the X-axis.
 - Right: Sets the pivot to the far right side.
- Y-Axis (Top to Bottom):
 - Top: Sets the pivot to the topmost point of the object's bounding box.
 - Centre: Aligns the pivot to the center vertically.

- Bottom: Sets the pivot to the lowest vertical point.
- Z-Axis (Front to Back):
 - Back: Aligns the pivot to the farthest point away from the camera.
 - Centre: Places the pivot at the Z-center of the object.
 - Front: Aligns the pivot to the point closest to the camera.

Clicking a node in this 3x3x3 cube updates the pivot to the corresponding logical point. The setting is reflected immediately in both the pivot and the Custom Pivot section.

Custom Pivot Section: This area includes numeric fields for fine-tuned manual control:

- X: Moves the pivot left or right relative to the object.
- Y: Moves the pivot up or down.
- Z: Moves the pivot forward or backward.

Users can enter values directly or use mouse-drag interactions for quick adjustments. These settings allow placing the pivot beyond the default bounding box for advanced layout needs.

Pivot Manipulation in Viewport: To manipulate the pivot directly in the viewport:

- Press the T key to activate Pivot Mode.
- Use mouse controls or animator tools to reposition the pivot interactively.

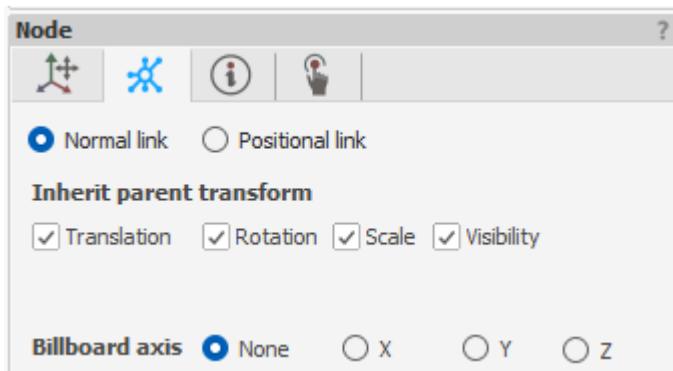
✓ **Usage Example:** To animate an object rotating from its bottom center, set the pivot to Centre, **Bottom, Centre**. For slide-in effects from the side, set the pivot to Left, Centre, Front and animate along the X-axis.

Tip: Combining Pivot Preset placement with animation gives users full control over motion dynamics without needing additional parent layers.

Node

The Node Tab is used to define hierarchical behavior and spatial relationships between objects in the scene. It allows objects to inherit transformations, align with each other using bounding-box logic, maintain face orientation relative to the camera, and define interactivity based on hover states. It is especially useful when setting up dynamic, rule-based object grouping, anchoring, or behavior-driven layouts.

Normal Link



Normal Link establishes a classic parent-child relationship between two objects. When applied:

- The child inherits position, rotation, and scale from the parent.
- All transformations on the parent directly affect the child in real time.

This is ideal for creating logical hierarchies—like attaching labels to a moving frame, grouping elements, or syncing transformations between related UI components.

 **Tip:** Use Normal Link when you want full transformation inheritance between objects.

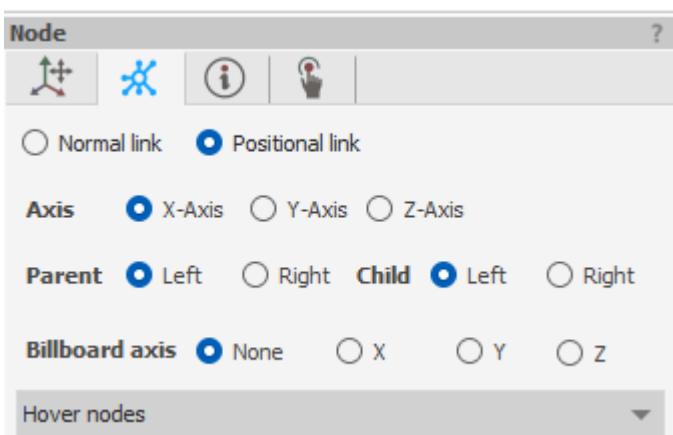
Inherit Parent Transform : This section gives you precise control over which transformations are inherited from the parent object. Each property is toggled individually:

- Translation: The child moves with the parent's position.
- Rotation: The child rotates relative to the parent.
- Scale: The child scales proportionally with the parent.
- Visibility: The child becomes hidden or shown based on the parent's visibility.

✓ Use Case: If you want a child to move with its parent but rotate independently, enable only Translation and disable Rotation.

 **Tip:** Fine-tune these toggles when mixing static and dynamic elements within a group.

Position Link



The Position Link feature is used to align and attach the position of one object relative to another along

a specified axis. It enables dynamic, rule-based positioning using object bounds such as left, right, top, bottom, near, and far—depending on the axis selected.

Understanding Object Bounds

Position linking operates using object bounds, which are the physical limits of a 3D object in space. These bounds define the left, right, top, bottom, near, and far extents used in relative alignment.

! Important Note: Groups do not have their own physical bounds. A group is an empty container, and its bounds are derived only from the objects it contains.

As a result, using a group as the parent or child in a position link will not yield expected results unless the group contains at least one visible object with defined bounds.

To enable Position Link functionality between grouped objects:

- You must manually parent objects by dragging one object under another in the scene hierarchy.
- This ensures that the Position Link calculations can access valid bounds for each object.

Axis-Based Linking

Position linking can be performed along any of the three primary axes: X, Y, and Z.

- X-Axis Link

- Parent Options:
 - Left - Uses the left boundary of the parent object
 - Right - Uses the right boundary of the parent object
- Child Options:
 - Left - Aligns the left boundary of the child
 - Right - Aligns the right boundary of the child

 **Example:** Align the left edge of the child object to the right edge of the parent object to position it directly beside it on the X-axis.

- Y-Axis Link

- Parent Options:
 - Bottom — Uses the bottom boundary of the parent object
 - Top — Uses the top boundary of the parent object
- Child Options:
 - Bottom — Aligns the bottom boundary of the child
 - Top — Aligns the top boundary of the child

 **Example:** Attach the top edge of a label to the bottom edge of a graphic panel.

- Z-Axis Link

- Parent Options:

- Near - Uses the near (closer to camera) boundary of the parent object
- Far - Uses the far (away from camera) boundary of the parent object
- Child Options:
 - Near - Aligns the near face of the child
 - Far - Aligns the far face of the child

 **Example:** Stack a background plane behind a foreground object using Z-axis Near/Far linking.

Usage Notes: Always ensure both parent and child objects have visible geometry with non-zero bounds. Do not use empty groups as the target of a position link unless they contain measurable child objects. Position links are non-destructive—they maintain relative positioning even when objects are resized dynamically or replaced.

Billboard Axis

The Billboard Axis setting determines how an object or texture faces the camera during rendering. This is useful for scenarios such as text labels, icons, or particles that should remain oriented toward the viewer.

This property is controlled using a set of radio buttons, with the following available options:

None

- No billboarding is applied.
- The object maintains its original orientation in 3D space, regardless of camera position.
- Use this mode when you want the object to behave like a standard 3D object.

X

- Enables billboarding around the X-axis only.
- The object rotates to face the camera horizontally, but maintains its vertical orientation.
- Commonly used for upright elements that need horizontal alignment with the viewer (e.g., standing banners, pillars).

Y

- Enables billboarding around the Y-axis only.
- The object rotates to face the camera vertically, while keeping horizontal orientation intact.
- Suitable for elements like text or flat icons that should rotate left/right to face the camera.

Z

- Enables billboarding around the Z-axis only.
- The object faces the camera while allowing rotation around the Z-axis (into/out of the screen).
- Typically used for flat sprites or effects that follow the viewer head-on.

Usage Tip: Choose the axis that defines which orientation should remain fixed. The object will rotate on the other axes to face the camera accordingly.

Hover Nodes

The Hover Nodes section allows users to define child objects that respond to mouse hover events on a parent object. This is useful for creating interactive effects where specific objects appear, animate, or trigger actions when a user hovers over another object.

Hover Nodes Panel

- A visible empty box labeled Hover Nodes appears in the object's properties panel.
- This panel is used to associate other objects that should respond when the user hovers over the parent object.

Usage Instructions

- Select the Parent Object in the hierarchy panel—the one that should detect the hover event.
- Drag one or more objects from the hierarchy pool into the Hover Nodes box.
 - These dragged objects become the hover responders.
 - They will now react based on the defined hover events of the parent.

 **Hover Nodes must be assigned while the intended parent object is selected.**

On Drag Hover Command : Once Hover Nodes are assigned, the parent object can have hover-related events configured using the On Drag Hover command. These include:

- Hover In: Triggered when the mouse pointer enters the boundary of the parent object.
- **Hover:** Continuously triggered while the mouse pointer remains within the object's bounds.
- **Hover Out:** Triggered when the mouse pointer exits the boundary of the parent object.

Each event can have a set of Actions (e.g., animations, property changes, visibility toggles) assigned to it via action sets or timeline events.

Display Properties

The Display Properties Tab provides advanced visibility and diagnostic controls for any selected object in the scene. These settings do not affect the geometry or animation of an object but control how it is rendered, previewed, and debugged in both the editor and the final output.

1. Object Lock: The Object Lock option is used to lock the transformation of an object. When enabled:

- The object cannot be moved, rotated, or scaled using the viewport controls.
- It remains fixed in position and size, preventing accidental manipulation.
- Useful for reference layers, UI frames, or layout grids.

 **Use Case: Lock background panels or guides so they don't shift during scene editing.**

2. Frustum Culling : Frustum Culling determines whether an object should be rendered based on its visibility inside the viewport (camera frustum).

- When enabled (checked), the object is only rendered if visible within the camera's viewing volume.
- This improves performance by skipping rendering of off-screen or out-of-bounds objects.
- Can be toggled:
 - Per object (in the Display tab)
 - From the Hierarchy via right-click
 - Globally using **Scenegraph Variable → Frustum Culling**

 **Tip:** Enable Frustum Culling for scene-intensive projects to reduce processing load

3. Render: The Render toggle determines if the object should be included in the final output.

- Checked: The object is visible in the final render and previews.
- Unchecked: The object is completely hidden from view and not processed during rendering.

This property can also be dynamically controlled using a Render Action, allowing objects to appear/disappear during timeline playback, Action Set triggers, or events.

 **Use Case:** Hide placeholder objects during runtime, or show overlays conditionally.

4. Motion Trail : The Motion Trail feature visualizes the movement of an object over time by generating a trailing effect based on past positions.

Parameters:

- Steps: Controls the number of positional frames used to build the trail.
 - ➤ Higher values produce smoother trails.
 - ➤ Lower values create a choppier, stylized trail.
- **Object Copies:** Defines how many ghosted copies of the object are used to visualize the trail.
 - ➤ More copies increase visual density.
 - ➤ Fewer copies give a sparse effect.
- Trail Transparency: A curve graph that controls the opacity of the trail over time.
 - Users can manipulate Bezier points to customize the transparency falloff.
 - Drag curve points vertically to increase/decrease transparency.
 - Use Alt + Click to insert new points.
 - Right-click a point to:
 - Set as Corner or Smooth
 - Delete the point if needed.

 **Tip:** Combine Motion Trail with Trail Transparency for dynamic ghosting of fast-moving objects.

5. Screen Info: The Screen Info section provides the object's position and size relative to the 2D screen space in the viewport. These values are dynamically calculated based on the camera view and screen resolution.

- X: Horizontal pixel coordinate of the object's position.
- Y: Vertical pixel coordinate of the object's position.
- Width: Visible width in pixels.
- Height: Visible height in pixels.

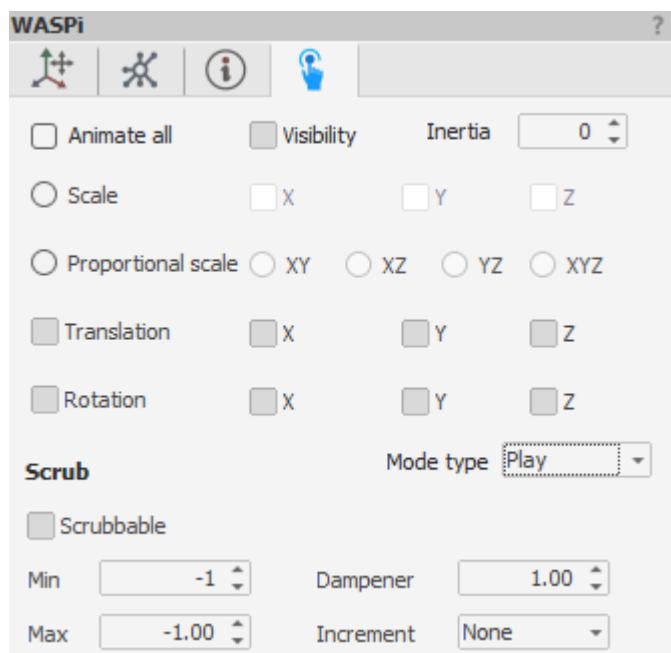
✓ **Use Case:** Use this data for aligning HUD elements, measuring relative screen area, or anchoring UI components.

6. Bounding Box Info: The Bounding Box Info defines the object's minimum and maximum extents in the 3D scene. This data is based on the object's actual geometry and is not affected by screen resolution or view.

- Min X: Left-most horizontal coordinate in world space.
- Min Y: Bottom-most vertical coordinate.
- Min Z: Closest depth (toward the camera).
- Max X: Right-most horizontal coordinate.
- Max Y: Top-most vertical coordinate.

WASPi

The Waspi Tab provides touch-interactive behavior, scrubbing control, and animation toggles for objects used in interactive or tool-driven scenes. These settings are especially relevant when working in touch-enabled environments using the Playout Engine with Tools Addin.



1. Animate All: The Animate All checkbox determines whether the selected object can be animated in a touch-enabled environment.
 - When enabled, it activates all transformation controls (Translate, Rotate, Scale) for the selected object.

- This setting is essential for interactive use cases where users touch or drag objects directly on-screen.
- Requires:
 - Tools Addin enabled in the Playout Engine
 - Scene to have the Tools Addin present

✓ Use Case: Enable **Animate All** for draggable cards, sliders, or on-screen elements with kinetic behavior.

2. Visibility:

Controls the runtime visibility of the object

- Checked: Object is visible and interactive in the scene.
- Unchecked: Object is hidden from both view and interaction

This is different from Display Properties > Render, as this option is intended for touch-driven visibility toggling.

3. Inertia:

Inertia defines the object's resistance to motion or interaction when being moved or transformed by touch input.

- High Inertia: More resistant to movement. Objects glide slowly and require stronger swipes to move.
- Low Inertia: Responds quickly to interaction, with minimal drag resistance.

Tip: Use higher inertia for heavy-feeling UI elements or lower inertia for responsive, snappy controls.

4. Scale and Proportional Scale:

The user can choose between Scale and Proportional Scale to determine how the object is resized:

- **a. Scale:** Allows axis-specific resizing of the object independently.
 - X: Change width without affecting height or depth.
 - Y: Change height independently.
 - Z: Modify depth independently.
- **b. Proportional Scale:** Maintains a locked ratio between axes for uniform scaling along multiple directions.
 - XY: Proportional scaling in horizontal and vertical directions.
 - XZ: Proportional scaling in width and depth.
 - YZ: Proportional scaling in height and depth.
 - XYZ: Scales all dimensions proportionally to maintain full 3D ratio.

5. Translation:

Controls the object's position adjustment in space via touch interactions

- X: Allows horizontal movement.
- Y: Allows vertical movement.
- Z: Allows depth movement (in/out of the screen).

These axes are independent and can be enabled or disabled based on the desired interaction model.

6. Rotation:

Enables axis-specific rotation:

- X: Rotates the object around the horizontal axis.
- Y: Rotates it around the vertical axis.
- Z: Rotates it around the forward/backward axis.

Tip: Combine rotation with inertia for smooth rotational dragging effects.

Scrub

The Scrub section enables parameter value animation independent of the scene timeline. Ideal for use cases like menu sliders, sliders for effects, or interactive dials.

Mode Type: The Mode setting defines how an object responds to animation triggers—either through the regular timeline or through user-driven interaction (mouse/touch). It determines whether the object follows predefined animations or reacts in real time based on external gestures or actions.

This option is available in the form of a dropdown with two choices:

Play

- Default Behavior: Objects are automatically set to Play mode upon creation.
- In Play mode, the object follows standard animation cues:
 - Controlled by keyframes in the Trackview
 - Can also be triggered using Action Sets
- Object behavior is time-based and not affected by user gestures.

✓ **Use Case:** Use Play mode for animated elements that follow a timeline, such as logo reveals, automatic sliders, or character intros.

Step

- Enables gesture-driven animation.
- The object responds to mouse drag or touch swipe interactions.
- Useful for scenarios like:
 - Carousels
 - Draggable sliders
 - Scrollable menus
- The object's animation is not linked to the Trackview timeline but is driven directly by user motion.

⇨ **Step Mode + Jump Action:** When touch or mouse interaction is not used, you can still control step-based animations using the Jump Action, allowing frame or state transitions programmatically or via a trigger (like button press).

Refer to Jump Action in the Action Controls documentation for more detail.

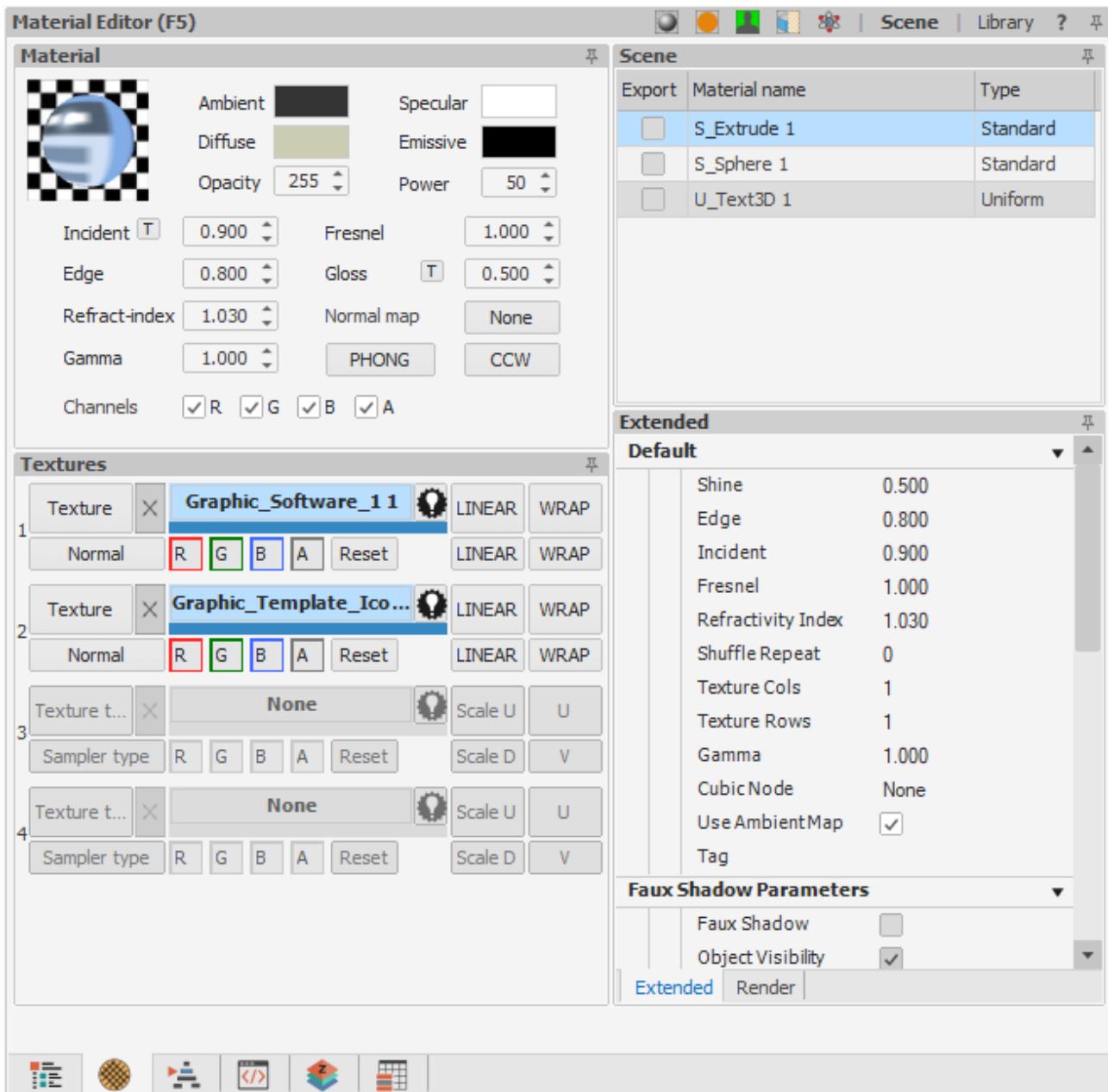
✓ **Use Case:** Interactive touchscreens, swipe-based product displays, or rotating 3D item previews.

❖ **Tip:** Switch to Step mode for responsive, interactive behavior. Use Play for scene-driven sequences.

- **Scrabbleable:** When checked, allows the object or parameter to be scrubbed interactively, independent of time.

- Users can scrub by touch/drag gestures.
- Value changes reflect in real time and respond immediately to user interaction.
- **Min:** Sets the starting value or minimum frame for the scrub range.
 - Defines the lower boundary of how far the scrub action can go.
 - Commonly used to reset position, opacity, or animation states.
- **Max:** Defines the maximum value or last frame for the scrub range.
 - Serves as the upper cap of the scrub range.
 - Used to bound animation like progress bars or object movement.
- **Dampener:** Controls the smoothness of the scrubbing transition.
 - A higher value slows down the value transition, making changes gradual.
 - A lower value results in sharp, snappy changes.
 - Helps create natural motion effects during rapid or aggressive user interaction.
 - ✓ **Use Case:** For analog knobs or scrubbing across media timelines, a higher dampener smooths fluctuations.
- **Increment Direction:** Sets the direction of incremental value change when scrubbing is applied.
- **Options:**
 - +X: Increment toward the right
 - -X: Decrement toward the left
 - +Y: Increment upward
 - -Y: Decrement downward
 - ❖ **Tip:** Match scrub direction with on-screen movement for intuitive UI experiences—such as a vertical menu slider using +Y.

Strandard Material



Material Editor (F5)

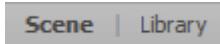
It has 5 types of **Materials**



- Standard Material: This Material type is used on 3D shapes. The Standard material provides a reflective surface to an object. A surface displaying single color usually reflects many colors. Standard materials use a three-color model (Ambient, Diffuse and Specular) to show color blending and reflective surface. Apart from color scheme, the visual appearance of the surface is determined by the reflective properties, transparency properties and texture layers. Also, Standard Material shows an effect of light and different lights can be assigned for different Materials. It can be used for creating most of the real world materials like Wood, Plastic, Steel, etc.

- Uniform: This Material type is used with 2D shapes. It is a non-reflective material and has no effect of light. The Diffuse color component provides the Fill color to the material. Multiple texture layers, Texture Types along with their blending options can be used with this material. Uniform material as four texture layers.
- Chroma: It provides real-time internal Chroma-keying. It allows blending between a live video texture source (shot on a blue/green background) and key out (make transparent) the background from this video texture in real-time. Chroma material contains two texture layers only. This helpful when the user wants to apply opacity map on the Chroma object.
- Matte: It is applied to the objects to make them invisible and block all objects behind it except the background. It has no color and texture layers. Matte Material can be used for selected objects using the Stencil-matte property of the object available in Parameter pool (F4).
- Particle: This Material type is used on particle object. By applying this material to a particle, you can make a particle change its color according to its life (as in frames), by defining the particle Start Color, Mid Color and End Color. Particle material can also use a grid based sprite and DDS volume texture to use the sprites as different particles of the particle object. The sprite texture can be divided using the Row and Columns properties in Extended Parameters -> Particle Texture Parameters.

The Material Pool window has two tabs:



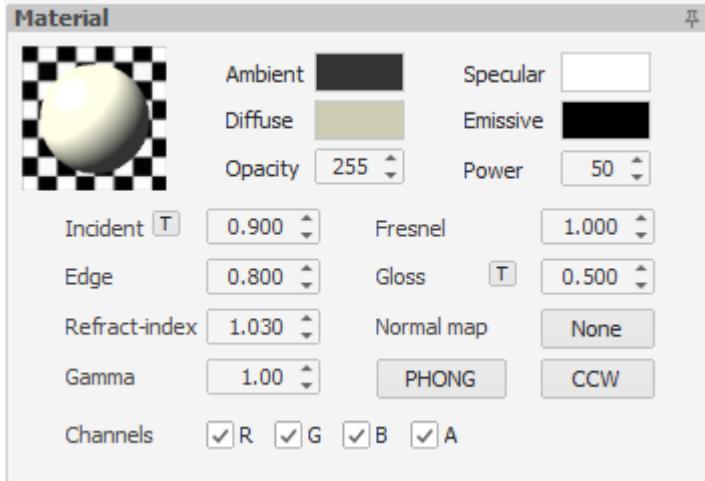
- **Scene** – Provides an interface to define & modify materials.
- **Library** – Has as a collection of materials kept as references or for repeated use on the user's local system. It enables users to create/delete folder, view materials in Large Icon/Details, etc.

Channels - The Channels section allows users to manage the RGBA (Red, Green, Blue, Alpha) output channels for a material. Each channel can be individually enabled or disabled based on the specific design requirements.

For instance, in Virtual Set environments where objects are intended to appear behind a chroma key, the Alpha Channel is typically disabled to ensure proper compositing. If transparency is needed, users can simply enable the Alpha channel by checking the corresponding option.

This feature offers precise control over how materials are rendered and composited within the scene.

1. Material

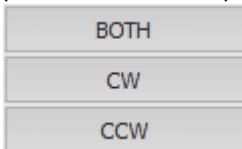


- **Ambient** : The base or inherent color of an object under uniform or ambient lighting conditions. It often represents the overall environment's lighting impact on object.
- **Preview** – Displays a preview of the material applied to a spherical object, allowing visualization of how the material appears when mapped onto an object.
- **Diffuse** : The inherent surface color or texture of an object that evenly scatters light, determining its appearance under direct lighting.
- **Specular** : The color of the highlight or shine on a surface caused by direct light sources, influencing the perceived material's reflective quality.
- **Emissive** : The color of materials or surfaces that emit light themselves, independently of external light sources.
- **Opacity** : The degree to which the color of an object or material is transparent or visible. Higher opacity means less transparency.
- **Power** : In the context of specular highlights, it determines the tightness or spread of the highlight, affecting its size and intensity.
- **Incident** : The angle at which light hits a surface and causes reflection within the 3D environment.
- **Fresnel** : The property determining how reflective a surface appears based on the viewing angle, contributing to more realistic material appearance.
- **Edge** : Refers to the characteristics or attributes of a surface's edges in terms of reflection properties.
- **Gloss** : The smoothness or shininess of a material, affecting how it scatters light and influences the perception of its surface.
- **Refract-index** : The index determining how much light bends when passing through a material, impacting its transparency and optical behavior.
- **Normal map** : A texture map used to add surface detail and simulate geometry on low-resolution models by manipulating surface normal.
- **Gamma** : A measure of brightness and contrast in digital imaging, affecting color accuracy and display.
- **Shading** is the process of determining the color or appearance of individual pixels or fragments in a 3D scene. It involves calculating how light interacts with surfaces, determining the color and brightness of each pixel on the screen.

There are three shading techniques used by Wasp3D.



- Flat Shading: Assigns a single color to each polygon, resulting in a faceted look.
- Gouraud Shading: Interpolates vertex colors across the surface of polygons, creating smoother transitions between colors.
- Phong Shading: Calculates lighting at each pixel by interpolating normal's across the surface, resulting in more accurate highlights and reflections.
Standard Materials are typically rendered in Phong Shading by default. However, utilizing Flat or Gouraud shading methods, which are more GPU-friendly, can help cut down render times or serve specific design needs.
- Cullmode: Determines which side(s) of polygons are not rendered, optimizing rendering performance and preventing unnecessary calculations for unseen faces.



- BOTH: This term might refer to rendering or display modes where both front-facing and back-facing polygons or surfaces are visible and rendered. Typically, in many 3D engines or software, you can choose to render both sides of polygons (front and back) or only the front-facing side for optimization purposes.
- CCW: Abbreviation for "Counter ClockWise," often used in graphics to define the order of vertices that determine the orientation of a polygon face.
- CW: Stands for "Clockwise." In 3D graphics, it refers to the order in which vertices of a polygon are arranged to define the face of that polygon. When vertices are arranged in a clockwise order, it determines the front-facing side of the polygon.

Scene

Scene		
Export	Material name	Type
<input type="checkbox"/>	S_Cube 1	Standard
<input type="checkbox"/>	S_Cylinder 1	Standard
<input type="checkbox"/>	S_N_Gon 1	Standard
<input type="checkbox"/>	S_Ring 1	Standard
<input type="checkbox"/>	S_Sphere 1	Standard

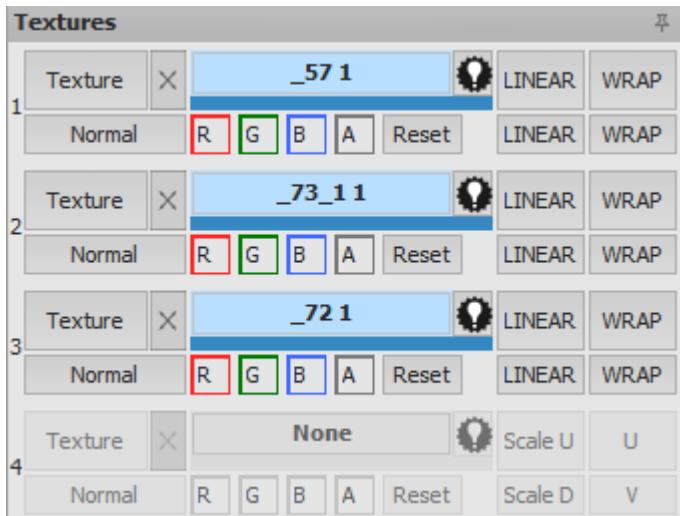
Export: When enabled, it allows selecting or applying the respective material from external applications.

Material name: Displays the name of the Materials

Type: Displays the type of Material.

User can Drag and Drop layers to change the layer order. Click on the Layer Number and Drag Up/Down using mouse. Press Shift+Drag to Swap between the selected texture layers.

- Textures

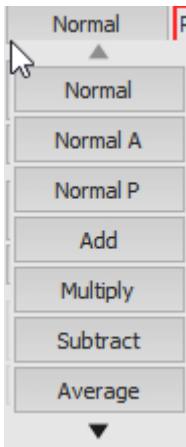


Texture Type defines the behavior of texture when mixed with material.



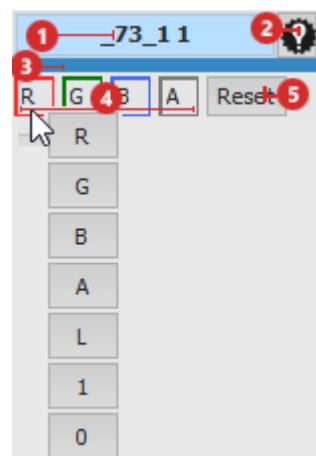
- Texture: The Texture option wraps a bitmap image onto an object's surface to create a visually realistic object. A bitmap consists of texture pixels or textures , where each texture provides the visual complexity which otherwise would have to be modeled. This advantage of lower polygon numbers helps reduce the rendering power while maintaining the frame-rate. X (X-Ray) option is available to enable X-Ray mode for texture when applied to a material of an object, the texture is scaled to the size of the viewport/render-area. The fill area of the object works as a window to reveal the texture.
- Opacity: Opacity is a kind of texture-map where the bitmap wrapped on the surface, is used to define the alpha (transparency) of the object surface. X option enables the selected texture to work in X-ray mode.
- Gradient: Gradient wipes the Luminance value of the selected channel.
- Mirror: The texture is mirrored and then tiled.

Normal - Filters



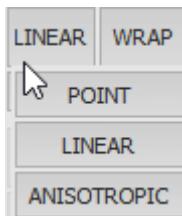
- Normal: The default mode, where no blending between texture maps occurs. Each texture pixel on the surface is painted as it appears in the original texture bitmap. Blending occurs between the base color and blend color based on the transparency of the texture pixels.
- Normal P: Works with Premultiplied Alpha, where the texture is blended with the layer behind it based on its alpha values. This mode assumes the colors in the texture have already been multiplied by the alpha, resulting in smoother blending, especially for semi-transparent textures.
- Normal A: In this mode, the alpha is multiplied separately, and colors are then blended based on their transparency. This method provides more control over how the colors interact with the layers behind, often leading to more precise edge blending.
- Add: This mode adds the base color to the blend color, resulting in a brighter color that moves towards white.
- Multiply: Multiplies the base color by the blend color, which results in a darker image. It is commonly used for shadow effects because multiplying any color by black results in black, while multiplying by white leaves the color unchanged.
- Subtract: Subtracts the blend color from the base color or vice versa, depending on which has a greater brightness value. This moves the result towards black.
- Average: Averages all the color components (RGBA) of the base and blend colors, resulting in a mid-tone between the two.
- Screen: Both the base and blend colors are inverted, multiplied by each other, and then the result is inverted again. Screening with black leaves the color unchanged, while screening with white results in white.
- Darken: Compares the base and blend colors and takes the darkest one as the output color.
- Lighten: Compares the base and blend colors and takes the lightest one as the output color.
- Exclusion: Produces a result where black and white return the same result, but colors in between become gray. This mode creates a "difference" effect but with lower contrast, and can be used to invert parts of an image.
- Overlay: A combination of multiply and screen modes, depending on the base color. If the base color is light, the blend color is screened. If the base color is dark, the blend color is multiplied.
- Hard Light: Uses the blend color as the base and either multiplies or screens the colors, depending on the blend color. If the blend color is lighter than 50% gray, the base is screened; if darker, it's multiplied.
- Soft Light: Similar to hard light but with a more subtle effect. It uses the blend color as the base and darkens or lightens the colors, adding a soft contrast to the image.

- Color Dodge: Brightens the base color to reflect the blend color by dividing the base by the inverse of the blend color. This creates a highlight effect.
- Color Burn: Darkens the base color to reflect the blend color by inverting, dividing, and inverting again. It creates a burn or shadow effect.
- Classic Color Dodge: Similar to Color Dodge but follows a more traditional color dodge algorithm, producing a smoother blend.
- Classic Color Burn: Similar to Color Burn but uses a traditional burn algorithm, giving a more controlled darkening effect.
- Linear Dodge: Similar to Color Dodge but uses linear interpolation, resulting in a smoother brightening effect.
- Echo: Adds shine or light to a texture. The output resembles soft light but with a more pronounced luminance effect.
- Glow: Similar to Echo but with the base and blend colors swapped. The result is closer to hard light, emphasizing highlights.
- Freeze: A variation of the Echo mode, where the base color is inverted, the blend color is inverted, and the result is inverted again, creating a cooling or frosted effect.
- Heat: Similar to Freeze but with the base and blend colors swapped, producing a warming effect rather than a cooling one.
- Interpolation: Similar to Average mode but with increased contrast between the base and blend colors.
- Stamp: The output is a blend of the base and blend colors, similar to Average, but works best with medium-toned colors, creating a stamped or embossed effect.
- Invert: Inverts the base color. Black becomes white, white becomes black, and other colors are converted to their complementary values.



- Texture Name: Displaying the name of the texture.
- Inherit Scene Light: Implies objects in a 3D scene adopting the lighting characteristics of the overall scene rather than having individualized lighting settings.
- Opacity Slider: Manage the transparency levels of the texture.
- RGBA
 1. R: Represents the Red color channel in digital graphics, defining the intensity of red in a color.
 2. G: Represents the Green color channel, determining the intensity of green in a color.
 3. B: Represents the Blue color channel, indicating the intensity of blue in a color.

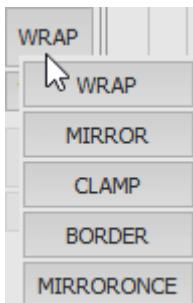
4. **A:** Denotes the Alpha channel, controlling transparency or opacity in addition to color.
 5. **L:** Represents the Luminance channel, often used to describe the brightness or lightness of a color.
 6. **1:** Often signifies the maximum value or full intensity in certain color channels, such as in some color models where 1 represents the highest value.
 7. **0:** Often represents the minimum value or absence of intensity in certain color channels, such as in some color models where 0 signifies no intensity or absence of color.
- **Reset:** Resetting the RGBA color values back to their default settings.
 - **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of he objects by specifying object. Tag as target.



Point: Color values of the closest adjacent pixel are taken to smoother the texture.

Linear: It is the default option & uses Bilinear Filtering in which a weighted average of four adjacent texture pixels is taken to create a single Texel.

Anisotropic: It takes into account the angle a primitive is rendered. Anisotropic compensates for distortion caused by the difference in angle between the texture polygon and the plane of the screen.



Wrap: Texture repeats n number of times in the U direction based on the value provided.

Mirror: The texture is mirrored and then tiled.

Clamp: Restricts the texture look up to the range 0 to 1. The unmapped area is mapped with the last pixel value.

Border: Generates a border for the texturemap.

Mirroronce: Similar to Mirror, the texture is mirrored only once and not tiled.

Extended

Displays extended set of parameters of the respective material.

Default		
	Shine	0.500
	Edge	0.800
	Incident	0.900
	Fresnel	1.000
	Refractivity Index	1.030
	Shuffle Repeat	0
	Texture Cols	1
	Texture Rows	1
	Gamma	1.000
	Cubic Node	None
	Use Ambient Map	<input checked="" type="checkbox"/>
	Tag	

Default

- **Shine:** Defines the strength/intensity of the specular color on the surface of the object.
- **Edge:** Sets the intensity of refection from the front of the object,
- Incident: Sets the intensity of refection from the front of the object.
- Fresnel: Sets the blend between the reflection Min and Max.
- Refractivity Index: Specifies the refractivity Index of the texture applied in the refraction texture mode.
- Roughness: Specifies the roughness of the material when a roughness mode is selected for a texture in the texture layer.
- Shuffle Repeat: Specifies the phase value for he texture cell animation. If it is 0, It would not repeat, when 1 it starts to shuffle across the texture cells.
- Texture Cols: Specifies the vertical division in the texture of the material. This can be used to animate the material as a sprite.
- Texture Row: Specifies the horizontal division in the texture of the material. This can be used to animate the material as a sprite.
- **Gamma:** Sets Gamma properties to Increase or decrease the color intensity of the material.
- Cubic Node: Reference node for local cubic map.
- Use Ambient Map: Applies Ambient light according to the Environment map.

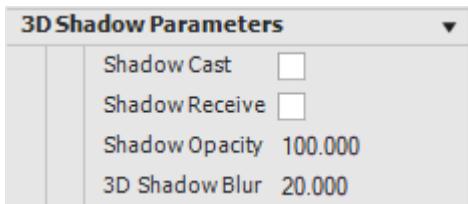
Faux Shadow Parameters

Faux Shadow Parameters	
	Faux Shadow <input type="checkbox"/>
	Object Visibility <input checked="" type="checkbox"/>
	Faux Shadow B... 0.000
	Blur Ratio 1.000
	Faux Shadow O... 0
	Faux Shadow A... 0
➤	Faux Shado... 0,0,0,200
	Shadow Depth 5.000

- **Faux Shadow:** Toggles visibility of shadow of the material
- Object Visibility: Toggles the visibility of the objects on which the selected material is applied.

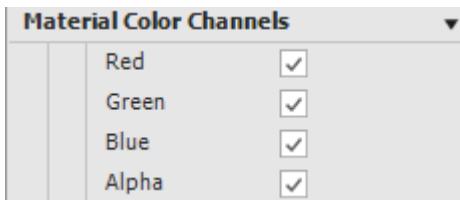
- **Faux Shadow Blur:** Sets the amount of blur of the faux shadow.
- **Blue Ratio:** Sets the scale of the blur of the faux shadow,
- **Faux Shadow Offset:** Sets the distance of shadow from the object shadow.
- **Faux Shadow Angel:** Sets the angle of refection in respect to the object.
- **Faux Shadow Color:** Sets the color and alpha of the shadow.
- **Shadow Depth:** Sets the shadow depth in respect to the object.

3D Shadow Parameters



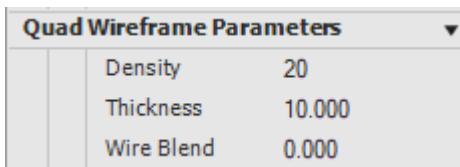
- **Shadow Cast:** Enables casting of shadow of the object to which the material is applied.
- **Shadow Receive:** Enables the object to receive 3D shadows on its surface to which the material applied.
- **Shadow Opacity:** Sets a normalized value for opacity of the casted shadow. Works on the receiver object.
- **3D Shadow Blue:** Sets a normalized value for the amount of blur of the casted shadow. Works on the receiver object.

Material Color Channels



- **Red:** When ON, renders the Red channel of the Material.
- **Green:** When ON, renders the Green channel of the Material.
- **Blue:** When ON, renders the Blue channel of the Material.
- **Alpha:** When ON, renders the Alpha channel of the Material.

Quad Wireframe Parameters



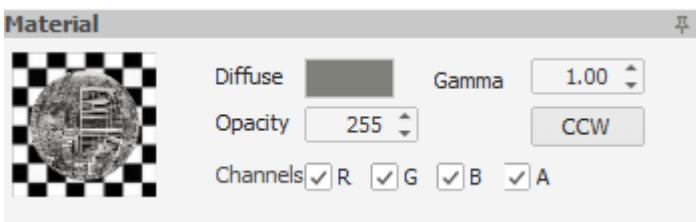
- **Density:** Sets the density of wire. Change fill mode to quad wire-frame to see the changes.
- **Thickness:** Sets the size of wire, Change fill mode to quad wire-frame to see the changes.
- **Wire Blend:** Sets the blend between fill and wire-frame of the object, change fill mode to quad wire-frame to see the changes,

Light Parameters

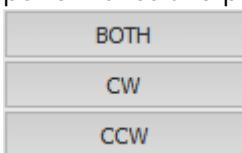


- Light Parameters: Specifies the light for the material. A separate light (other than default light) can be specified to affect the material.
- Sampler Type: Provides various texture blending modes, which can be used to blend between textures applied in different layers of the material.
- Render States: Provides options like fill mode, cull mode, point size and object blending parameters. Object blending works on the entire object to which the respective material is applied.

2. Material - Uniform

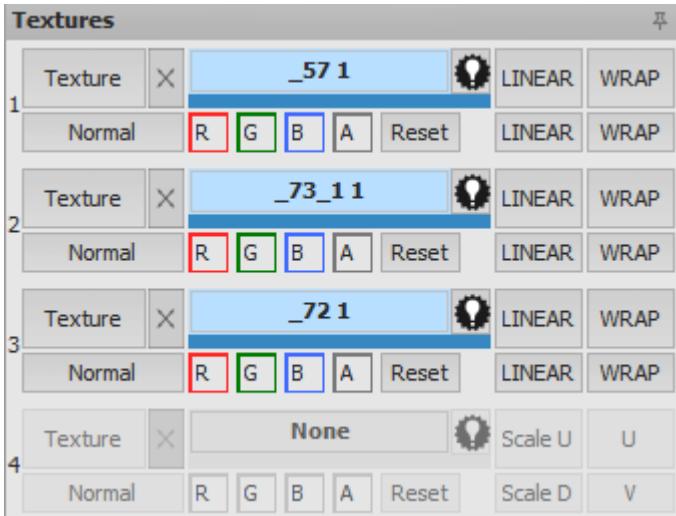


- Diffuse : The inherent surface color or texture of an object that evenly scatters light, determining its appearance under direct lighting.
- Gamma : A measure of brightness and contrast in digital imaging, affecting color accuracy and display.
- Opacity : The degree to which the color of an object or material is transparent or visible. Higher opacity means less transparency.
- Cullmode: Determines which side(s) of polygons are not rendered, optimizing rendering performance and preventing unnecessary calculations for unseen faces.



- BOTH: This term might refer to rendering or display modes where both front-facing and back-facing polygons or surfaces are visible and rendered. Typically, in many 3D engines or software, you can choose to render both sides of polygons (front and back) or only the front-facing side for optimization purposes.
- CCW: Abbreviation for "Counter ClockWise," often used in graphics to define the order of vertices that determine the orientation of a polygon face.
- CW: Stands for "Clockwise." In 3D graphics, it refers to the order in which vertices of a polygon are arranged to define the face of that polygon. When vertices are arranged in a clockwise order, it determines the front-facing side of the polygon.

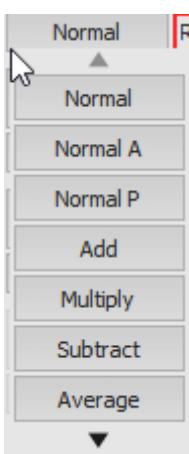
- Textures



Texture Type defines the behavior of texture when mixed with material.

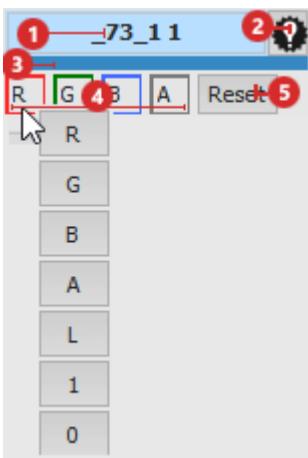


- Texture: The Texture option wraps a bitmap image onto an object's surface to create a visually realistic object. A bitmap consists of texture pixels or textures , where each texture provides the visual complexity which otherwise would have to be modeled. This advantage of lower polygon numbers helps reduce the rendering power while maintaining the frame-rate. X (X-Ray) option is available to enable X-Ray mode for texture when applied to a material of an object, the texture is scaled to the size of the viewport/render-area. The fill area of the object works as a window to reveal the texture.
- Opacity: Opacity is a kind of texture-map where the bitmap wrapped on the surface, is used to define the alpha (transparency) of the object surface. X option enables the selected texture to work in X-ray mode.
- Gradient: Gradient wipes the Luminance value of the selected channel.
- Mirror: The texture is mirrored and then tiled.
- Normal - Filters

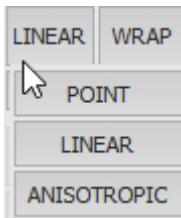


- Normal: The default mode, where no blending between texture maps occurs. Each texture pixel on the surface is painted as it appears in the original texture bitmap. Blending occurs between the base color and blend color based on the transparency of the texture pixels.
- Normal P: Works with Premultiplied Alpha, where the texture is blended with the layer behind it based on its alpha values. This mode assumes the colors in the texture have already been multiplied by the alpha, resulting in smoother blending, especially for semi-transparent textures.
- Normal A: In this mode, the alpha is multiplied separately, and colors are then blended based on their transparency. This method provides more control over how the colors interact with the layers behind, often leading to more precise edge blending.
- Add: This mode adds the base color to the blend color, resulting in a brighter color that moves towards white.
- Multiply: Multiplies the base color by the blend color, which results in a darker image. It is commonly used for shadow effects because multiplying any color by black results in black, while multiplying by white leaves the color unchanged.
- Subtract: Subtracts the blend color from the base color or vice versa, depending on which has a greater brightness value. This moves the result towards black.
- Average: Averages all the color components (RGBA) of the base and blend colors, resulting in a mid-tone between the two.
- Screen: Both the base and blend colors are inverted, multiplied by each other, and then the result is inverted again. Screening with black leaves the color unchanged, while screening with white results in white.
- Darken: Compares the base and blend colors and takes the darkest one as the output color.
- Lighten: Compares the base and blend colors and takes the lightest one as the output color.
- Exclusion: Produces a result where black and white return the same result, but colors in between become gray. This mode creates a "difference" effect but with lower contrast, and can be used to invert parts of an image.
- Overlay: A combination of multiply and screen modes, depending on the base color. If the base color is light, the blend color is screened. If the base color is dark, the blend color is multiplied.
- Hard Light: Uses the blend color as the base and either multiplies or screens the colors, depending on the blend color. If the blend color is lighter than 50% gray, the base is screened; if darker, it's multiplied.
- Soft Light: Similar to hard light but with a more subtle effect. It uses the blend color as the base and darkens or lightens the colors, adding a soft contrast to the image.
- Color Dodge: Brightens the base color to reflect the blend color by dividing the base by the inverse of the blend color. This creates a highlight effect.
- Color Burn: Darkens the base color to reflect the blend color by inverting, dividing, and inverting again. It creates a burn or shadow effect.
- Classic Color Dodge: Similar to Color Dodge but follows a more traditional color dodge algorithm, producing a smoother blend.
- Classic Color Burn: Similar to Color Burn but uses a traditional burn algorithm, giving a more controlled darkening effect.
- Linear Dodge: Similar to Color Dodge but uses linear interpolation, resulting in a smoother brightening effect.
- Echo: Adds shine or light to a texture. The output resembles soft light but with a more pronounced luminance effect.

- Glow: Similar to Echo but with the base and blend colors swapped. The result is closer to hard light, emphasizing highlights.
- Freeze: A variation of the Echo mode, where the base color is inverted, the blend color is inverted, and the result is inverted again, creating a cooling or frosted effect.
- Heat: Similar to Freeze but with the base and blend colors swapped, producing a warming effect rather than a cooling one.
- Interpolation: Similar to Average mode but with increased contrast between the base and blend colors.
- Stamp: The output is a blend of the base and blend colors, similar to Average, but works best with medium-toned colors, creating a stamped or embossed effect.
- Invert: Inverts the base color. Black becomes white, white becomes black, and other colors are converted to their complementary values.
-



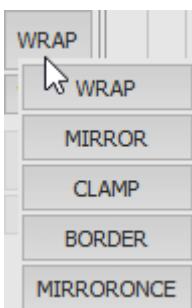
- Texture Name: Displaying the name of the texture.
- Inherit Scene Light: Implies objects in a 3D scene adopting the lighting characteristics of the overall scene rather than having individualized lighting settings.
- Opacity Slider: Manage the transparency levels of the texture.
- RGBA
 1. R: Represents the Red color channel in digital graphics, defining the intensity of red in a color.
 2. G: Represents the Green color channel, determining the intensity of green in a color.
 3. B: Represents the Blue color channel, indicating the intensity of blue in a color.
 4. A: Denotes the Alpha channel, controlling transparency or opacity in addition to color.
 5. **L:** Represents the Luminance channel, often used to describe the brightness or lightness of a color.
 6. **1:** Often signifies the maximum value or full intensity in certain color channels, such as in some color models where 1 represents the highest value.
 7. **0:** Often represents the minimum value or absence of intensity in certain color channels, such as in some color models where 0 signifies no intensity or absence of color.
- **Reset:** Resetting the RGBA color values back to their default settings.
- **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of he objects by specifying object. Tag as target.



Point: Color values of the closest adjacent pixel are taken to smoother the texture.

Linear: It is the default option & uses Bilinear Filtering in which a weighted average of four adjacent texture pixels is taken to create a single Texel.

Anisotropic: It takes into account the angle a primitive is rendered. Anisotropic compensates for distortion caused by the difference in angle between the texture polygon and the plane of the screen.



Wrap: Texture repeats n number of times in the U direction based on the value provided.

Mirror: The texture is mirrored and then tiled.

Clamp: Restricts the texture look up to the range 0 to 1. The unmapped area is mapped with the last pixel value.

Border: Generates a border for the texturemap.

Mirroronce: Similar to Mirror, the texture is mirrored only once and not tiled.

Extended

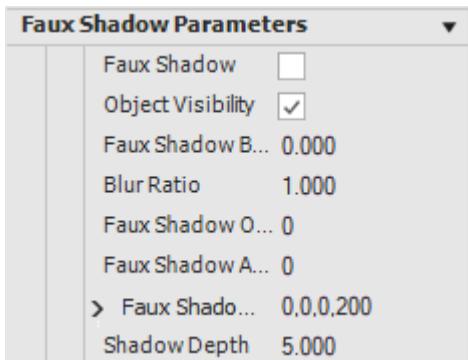
Displays extended set of parameters of the respective material.

Default		
	Shine	0.500
	Edge	0.800
	Incident	0.900
	Fresnel	1.000
	Refractivity Index	1.030
	Shuffle Repeat	0
	Texture Cols	1
	Texture Rows	1
	Gamma	1.000
	Cubic Node	None
	Use Ambient Map	<input checked="" type="checkbox"/>
	Tag	

Default

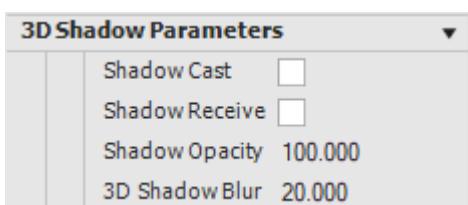
- **Shine:** Defines the strength/intensity of the specular color on the surface of the object.
- **Edge:** Sets the intensity of refection from the front of the object,
- Incident: Sets the intensity of refection from the front of the object.
- Fresnel: Sets the blend between the reflection Min and Max.
- Refractivity Index: Specifies the refractivity Index of the texture applied in the refraction texture mode.
- Roughness: Specifies the roughness of the material when a roughness mode is selected for a texture in the texture layer.
- Shuffle Repeat: Specifies the phase value for he texture cell animation. If it is 0, It would not repeat, when 1 it starts to shuffle across the texture cells.
- Texture Cols: Specifies the vertical division in the texture of the material. This can be used to animate the material as a sprite.
- Texture Row: Specifies the horizontal division in the texture of the material. This can be used to animate the material as a sprite.
- **Gamma:** Sets Gamma properties to Increase or decrease the color intensity of the material.
- Cubic Node: Reference node for local cubic map.
- Use Ambient Map: Applies Ambient light according to the Environment map.

Faux Shadow Parameters



- **Faux Shadow:** Toggles visibility of shadow of the material
- Object Visibility: Toggles the visibility of the objects on which the selected material is applied.
- Faux Shadow Blur: Sets the amount of blur of the faux shadow.
- **Blue Ratio:** Sets the scale of the blur of the faux shadow,
- **Faux Shadow Offset:** Sets the distance of shadow from the object shadow.
- **Faux Shadow Angel:** Sets the angle of refection in respect to the object.
- **Faux Shadow Color:** Sets the color and alpha of the shadow.
- **Shadow Depth:** Sets he Shadow depth in respect to the object.

3D Shadow Parameters



- **Shadow Cast:** Enables casting of shadow of the object to which the material is applied.
- **Shadow Receive:** Enables the object to receive 3D shadows on its surface to which the material applied.
- **Shadow Opacity:** Sets a normalized value for opacity of the casted shadow. Works on the receiver object.
- **3D Shadow Blue:** Sets a normalized value for the amount of blur of the casted shadow. Works on the receiver object.

Material Color Channels

Material Color Channels		
	Red	<input checked="" type="checkbox"/>
	Green	<input checked="" type="checkbox"/>
	Blue	<input checked="" type="checkbox"/>
	Alpha	<input checked="" type="checkbox"/>

- **Red:** When ON, renders the Red channel of the Material.
- **Green:** When ON, renders the Green channel of the Material.
- **Blue:** When ON, renders the Blue channel of the Material.
- **Alpha:** When ON, renders the Alpha channel of the Material.

Quad Wireframe Parameters

Quad Wireframe Parameters		
	Density	20
	Thickness	10.000
	Wire Blend	0.000

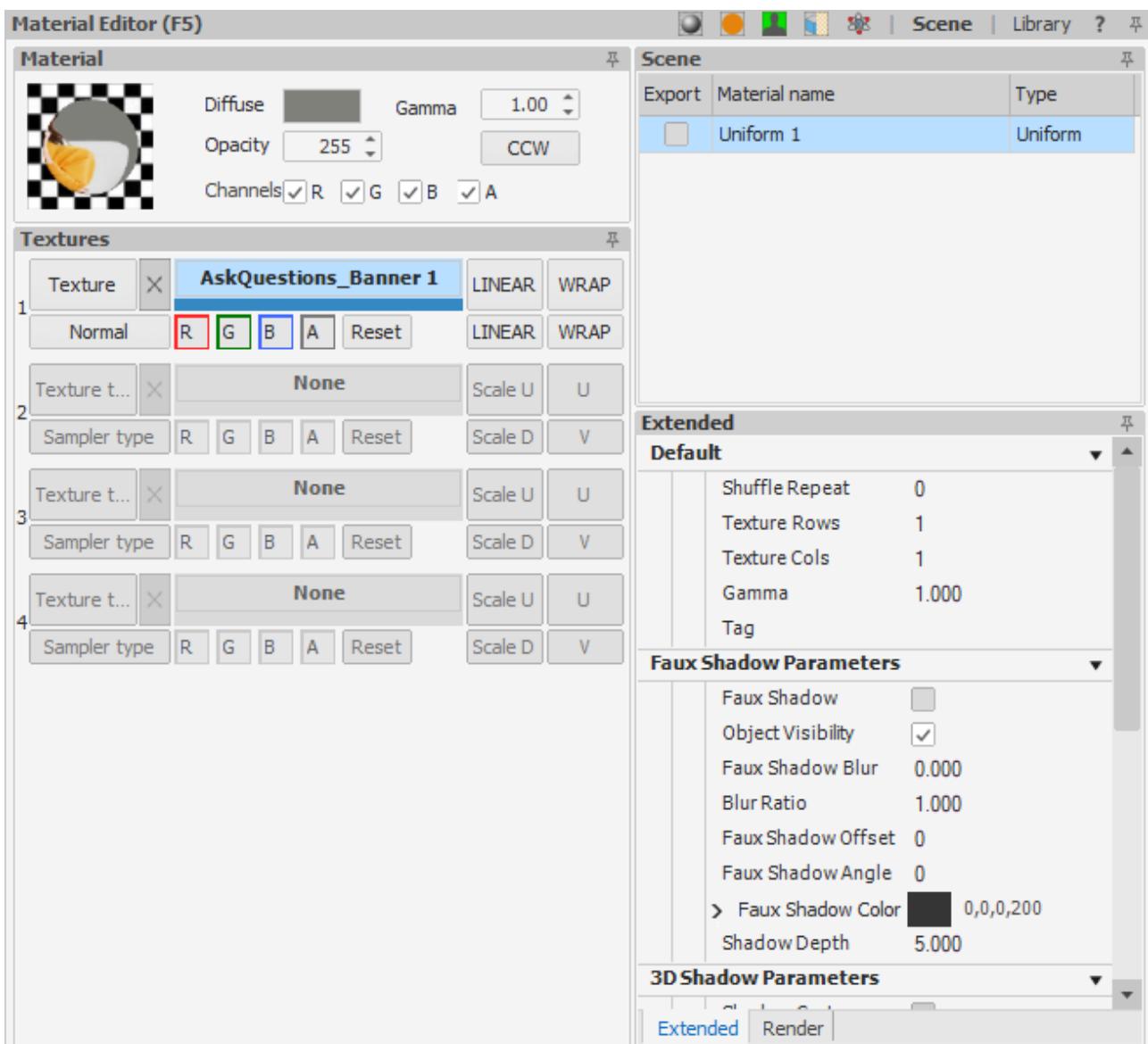
- **Density:** Sets the density of wire. Change fill mode to quad wire-frame to see the changes.
- **Thickness:** Sets the size of wire, Change fill mode to quad wire-frame to see the changes.
- **Wire Blend:** Sets the blend between fill and wire-frame of the object, change fill mode to quad wire-frame to see the changes,

Light Parameters

Light Parameters		
	Light	Directional Light 0

- **Light Parameters:** Specifies the light for the material. A separate light(other than default light) can be specified to affect the material.
- **Sampler Type:** Provides various texture blending modes, which can be used to blend between textures applied in different layers of the material.
- **Render States:** Provides options like fill mode, cull mode, point size and object blending parameters. Object blending works on the entire object to which the respective material is applied.

Uniform Material



Material Editor (F5)

It has 5 types of **Materials**

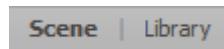


- **Standard:** This Material type is used on 3D shapes. The Standard material provides a reflective surface to an object. A surface displaying single color usually reflects many colors. Standard materials use a three-color model (Ambient, Diffuse and Specular) to show color blending and reflective surface. Apart from color scheme, the visual appearance of the surface is determined by the reflective properties, transparency properties and texture layers. Also, Standard Material shows an effect of light and different lights can be assigned for different Materials. It can be used for creating most of the real world materials like Wood, Plastic, Steel, etc.
- **Uniform:** This Material type is used with 2D shapes. It is a non-reflective material and has no effect of light. The Diffuse color component provides the Fill color to the material. Multiple

texture layers, Texture Types along with their blending options can be used with this material. Uniform material as four texture layers.

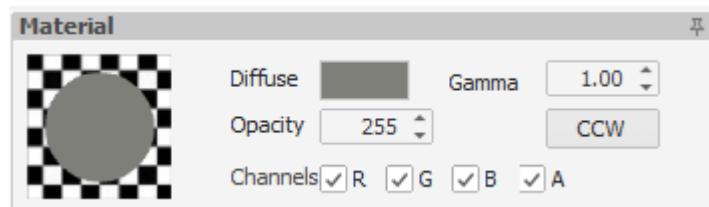
- Chroma: It provides real-time internal Chroma-keying. It allows blending between a live video texture source (shot on a blue/green background) and key out (make transparent) the background from this video texture in real-time. Chroma material contains two texture layers only. This helpful when the user wants to apply opacity map on the Chroma object.
- Matte: It is applied to the objects to make them invisible and block all objects behind it except the background. It has no color and texture layers. Matte Material can be used for selected objects using the Stencil-matte property of the object available in Parameter pool (F4).
- Particle: This Material type is used on particle object. By applying this material to a particle, you can make a particle change its color according to its life (as in frames), by defining the particle Start Color, Mid Color and End Color. Particle material can also use a grid based sprite and DDS volume texture to use the sprites as different particles of the particle object. The sprite texture can be divided using the Row and Columns properties in Extended Parameters -> Particle Texture Parameters.

The Material Pool window has two tabs:

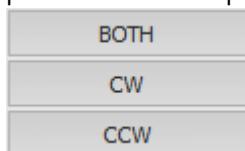


- **Scene** – Provides an interface to define & modify materials.
- **Library** – Has as a collection of materials kept as references or for repeated use on the user's local system. It enables users to create/delete folder, view materials in Large Icon/Details, etc.

1. Uniform Material



- Diffuse : The inherent surface color or texture of an object that evenly scatters light, determining its appearance under direct lighting.
- Gamma : A measure of brightness and contrast in digital imaging, affecting color accuracy and display.
- Opacity : The degree to which the color of an object or material is transparent or visible. Higher opacity means less transparency.
- Cullmode: Determines which side(s) of polygons are not rendered, optimizing rendering performance and preventing unnecessary calculations for unseen faces.



- BOTH: This term might refer to rendering or display modes where both front-facing and back-facing polygons or surfaces are visible and rendered. Typically, in many 3D engines or

software, you can choose to render both sides of polygons (front and back) or only the front-facing side for optimization purposes.

- CCW: Abbreviation for "Counter ClockWise," often used in graphics to define the order of vertices that determine the orientation of a polygon face.
- CW: Stands for "Clockwise." In 3D graphics, it refers to the order in which vertices of a polygon are arranged to define the face of that polygon. When vertices are arranged in a clockwise order, it determines the front-facing side of the polygon.
- Channels : The Channels section allows users to manage the RGBA (Red, Green, Blue, Alpha) output channels for a material. Each channel can be individually enabled or disabled based on the specific design requirements.

For instance, in Virtual Set environments where objects are intended to appear behind a chroma key, the Alpha Channel is typically disabled to ensure proper compositing. If transparency is needed, users can simply enable the Alpha channel by checking the corresponding option.

This feature offers precise control over how materials are rendered and composited within the scene.

Scene

Scene		
Export	Material name	Type
<input type="checkbox"/>	S_Cube 1	Standard
<input type="checkbox"/>	S_Cylinder 1	Standard
<input checked="" type="checkbox"/>	S_N_Gon 1	Standard
<input type="checkbox"/>	S_Ring 1	Standard
<input type="checkbox"/>	S_Sphere 1	Standard

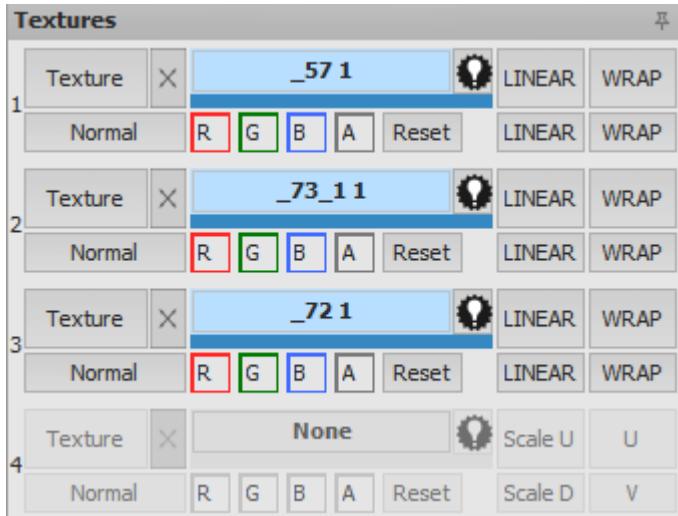
Export: When enabled, it allows selecting or applying the respective material from external applications.

Material name: Displays the name of the Materials

Type: Displays the type of Material.

User can Drag and Drop layers to change the layer order. Click on the Layer Number and Drag Up/ Down using mouse. Press Shift+Drag to Swap between the selected texture layers.

- Textures

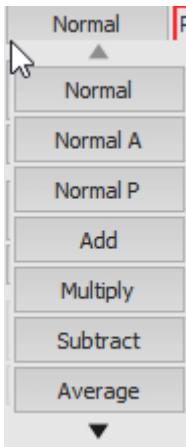


Texture Type defines the behavior of texture when mixed with material.



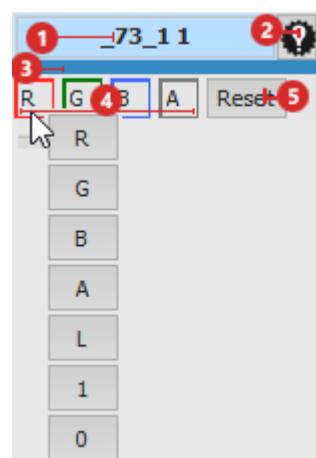
- Texture: The Texture option wraps a bitmap image onto an object's surface to create a visually realistic object. A bitmap consists of texture pixels or textures, where each texture provides the visual complexity which otherwise would have to be modeled. This advantage of lower polygon numbers helps reduce the rendering power while maintaining the frame-rate. X (X-Ray) option is available to enable X-Ray mode for texture when applied to a material of an object, the texture is scaled to the size of the viewport/render-area. The fill area of the object works as a window to reveal the texture.
- Opacity: Opacity is a kind of texture-map where the bitmap wrapped on the surface, is used to define the alpha (transparency) of the object surface. X option enables the selected texture to work in X-ray mode.
- Gradient: Gradient wipes the Luminance value of the selected channel.
- Mirror: The texture is mirrored and then tiled.

Normal - Filters



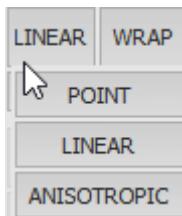
- Normal: The default mode, where no blending between texture maps occurs. Each texture pixel on the surface is painted as it appears in the original texture bitmap. Blending occurs between the base color and blend color based on the transparency of the texture pixels.
- Normal P: Works with Premultiplied Alpha, where the texture is blended with the layer behind it based on its alpha values. This mode assumes the colors in the texture have already been multiplied by the alpha, resulting in smoother blending, especially for semi-transparent textures.
- Normal A: In this mode, the alpha is multiplied separately, and colors are then blended based on their transparency. This method provides more control over how the colors interact with the layers behind, often leading to more precise edge blending.
- Add: This mode adds the base color to the blend color, resulting in a brighter color that moves towards white.
- Multiply: Multiplies the base color by the blend color, which results in a darker image. It is commonly used for shadow effects because multiplying any color by black results in black, while multiplying by white leaves the color unchanged.
- Subtract: Subtracts the blend color from the base color or vice versa, depending on which has a greater brightness value. This moves the result towards black.
- Average: Averages all the color components (RGBA) of the base and blend colors, resulting in a mid-tone between the two.
- Screen: Both the base and blend colors are inverted, multiplied by each other, and then the result is inverted again. Screening with black leaves the color unchanged, while screening with white results in white.
- Darken: Compares the base and blend colors and takes the darkest one as the output color.
- Lighten: Compares the base and blend colors and takes the lightest one as the output color.
- Exclusion: Produces a result where black and white return the same result, but colors in between become gray. This mode creates a "difference" effect but with lower contrast, and can be used to invert parts of an image.
- Overlay: A combination of multiply and screen modes, depending on the base color. If the base color is light, the blend color is screened. If the base color is dark, the blend color is multiplied.
- Hard Light: Uses the blend color as the base and either multiplies or screens the colors, depending on the blend color. If the blend color is lighter than 50% gray, the base is screened; if darker, it's multiplied.
- Soft Light: Similar to hard light but with a more subtle effect. It uses the blend color as the base and darkens or lightens the colors, adding a soft contrast to the image.

- Color Dodge: Brightens the base color to reflect the blend color by dividing the base by the inverse of the blend color. This creates a highlight effect.
- Color Burn: Darkens the base color to reflect the blend color by inverting, dividing, and inverting again. It creates a burn or shadow effect.
- Classic Color Dodge: Similar to Color Dodge but follows a more traditional color dodge algorithm, producing a smoother blend.
- Classic Color Burn: Similar to Color Burn but uses a traditional burn algorithm, giving a more controlled darkening effect.
- Linear Dodge: Similar to Color Dodge but uses linear interpolation, resulting in a smoother brightening effect.
- Echo: Adds shine or light to a texture. The output resembles soft light but with a more pronounced luminance effect.
- Glow: Similar to Echo but with the base and blend colors swapped. The result is closer to hard light, emphasizing highlights.
- Freeze: A variation of the Echo mode, where the base color is inverted, the blend color is inverted, and the result is inverted again, creating a cooling or frosted effect.
- Heat: Similar to Freeze but with the base and blend colors swapped, producing a warming effect rather than a cooling one.
- Interpolation: Similar to Average mode but with increased contrast between the base and blend colors.
- Stamp: The output is a blend of the base and blend colors, similar to Average, but works best with medium-toned colors, creating a stamped or embossed effect.
- Invert: Inverts the base color. Black becomes white, white becomes black, and other colors are converted to their complementary values.



- Texture Name: Displaying the name of the texture.
- Inherit Scene Light: Implies objects in a 3D scene adopting the lighting characteristics of the overall scene rather than having individualized lighting settings.
- Opacity Slider: Manage the transparency levels of the texture.
- RGBA
 1. R: Represents the Red color channel in digital graphics, defining the intensity of red in a color.
 2. G: Represents the Green color channel, determining the intensity of green in a color.
 3. B: Represents the Blue color channel, indicating the intensity of blue in a color.

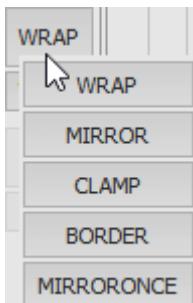
4. **A:** Denotes the Alpha channel, controlling transparency or opacity in addition to color.
 5. **L:** Represents the Luminance channel, often used to describe the brightness or lightness of a color.
 6. **1:** Often signifies the maximum value or full intensity in certain color channels, such as in some color models where 1 represents the highest value.
 7. **0:** Often represents the minimum value or absence of intensity in certain color channels, such as in some color models where 0 signifies no intensity or absence of color.
- **Reset:** Resetting the RGBA color values back to their default settings.
 - **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of he objects by specifying object. Tag as target.



Point: Color values of the closest adjacent pixel are taken to smoother the texture.

Linear: It is the default option & uses Bilinear Filtering in which a weighted average of four adjacent texture pixels is taken to create a single Texel.

Anisotropic: It takes into account the angle a primitive is rendered. Anisotropic compensates for distortion caused by the difference in angle between the texture polygon and the plane of the screen.



Wrap: Texture repeats n number of times in the U direction based on the value provided.

Mirror: The texture is mirrored and then tiled.

Clamp: Restricts the texture look up to the range 0 to 1. The unmapped area is mapped with the last pixel value.

Border: Generates a border for the texturemap.

Mirroronce: Similar to Mirror, the texture is mirrored only once and not tiled.

Extended

Displays extended set of parameters of the respective material.

Extended		
Default		
	Shuffle Repeat	0
	Texture Rows	1
	Texture Cols	1
	Gamma	1.000
	Tag	
Faux Shadow Parameters		
	Faux Shadow	<input type="checkbox"/>
	Object Visibility	<input checked="" type="checkbox"/>
	Faux Shadow Blur	0.000
	Blur Ratio	1.000
	Faux Shadow Offset	0
	Faux Shadow Angle	0
	➤ Faux Shadow Color	<input type="color"/> 0,0,0,200
	Shadow Depth	5.000
3D Shadow Parameters		
	Shadow Cast	<input type="checkbox"/>
	Shadow Receive	<input type="checkbox"/>
	Shadow Opacity	70.000
	3D Shadow Blur	20.000
Material Color Channels		
	Red	<input checked="" type="checkbox"/>
	Green	<input checked="" type="checkbox"/>
	Blue	<input checked="" type="checkbox"/>
	Alpha	<input checked="" type="checkbox"/>
Quad Wireframe Parameters		
	Density	20
	Thickness	10.000
	Wire Blend	0.000
Light Parameters		
	Light	Directional Light 0

Default

- **Shuffle Repeat:** Specifies the phase value for the texture cell animation. If it is 0, It would not repeat, when 1 it starts to shuffle across the texture cells.
- **Texture Row:** Specifies the horizontal division in the texture of the material. This can be used to animate the material as a sprite.
- **Texture Cols:** Specifies the vertical division in the texture of the material. This can be used to animate the material as a sprite.
- **Gamma:** Sets Gamma properties to Increase or decrease the color intensity of the material.
- **Tag:** Serves as a variable place holder to hold values for the object. User can specify object names in Tag and wire the equivalent properties of the objects by specifying Object,Tag as Target.

Faux Shadow Parameters

- **Faux Shadow:** Toggles visibility of shadow of the material
- **Object Visibility:** Toggles the visibility of the objects on which the selected material is applied.

- **Faux Shadow Blur:** Sets the amount of blur of the faux shadow.
- **Blue Ratio:** Sets the scale of the blur of the faux shadow,
- **Faux Shadow Offset:** Sets the distance of shadow from the object shadow.
- **Faux Shadow Angel:** Sets the angle of refection in respect to the object.
- **Faux Shadow Color:** Sets the color and alpha of the shadow.
- **Shadow Depth:** Sets the shadow depth in respect to the object.

3D Shadow Parameters

- **Shadow Cast:** Enables casting of shadow of the object to which the material is applied.
- **Shadow Receive:** Enables the object to receive 3D shadows on its surface to which the material applied.
- **Shadow Opacity:** Sets a normalized value for opacity of the casted shadow. Works on the receiver object.
- **3D Shadow Blue:** Sets a normalized value for the amount of blur of the casted shadow. Works on the receiver object.

Material Color Channels

- **Red:** When ON, renders the Red channel of the Material.
- **Green:** When ON, renders the Green channel of the Material.
- **Blue:** When ON, renders the Blue channel of the Material.
- **Alpha:** When ON, renders the Alpha channel of the Material.

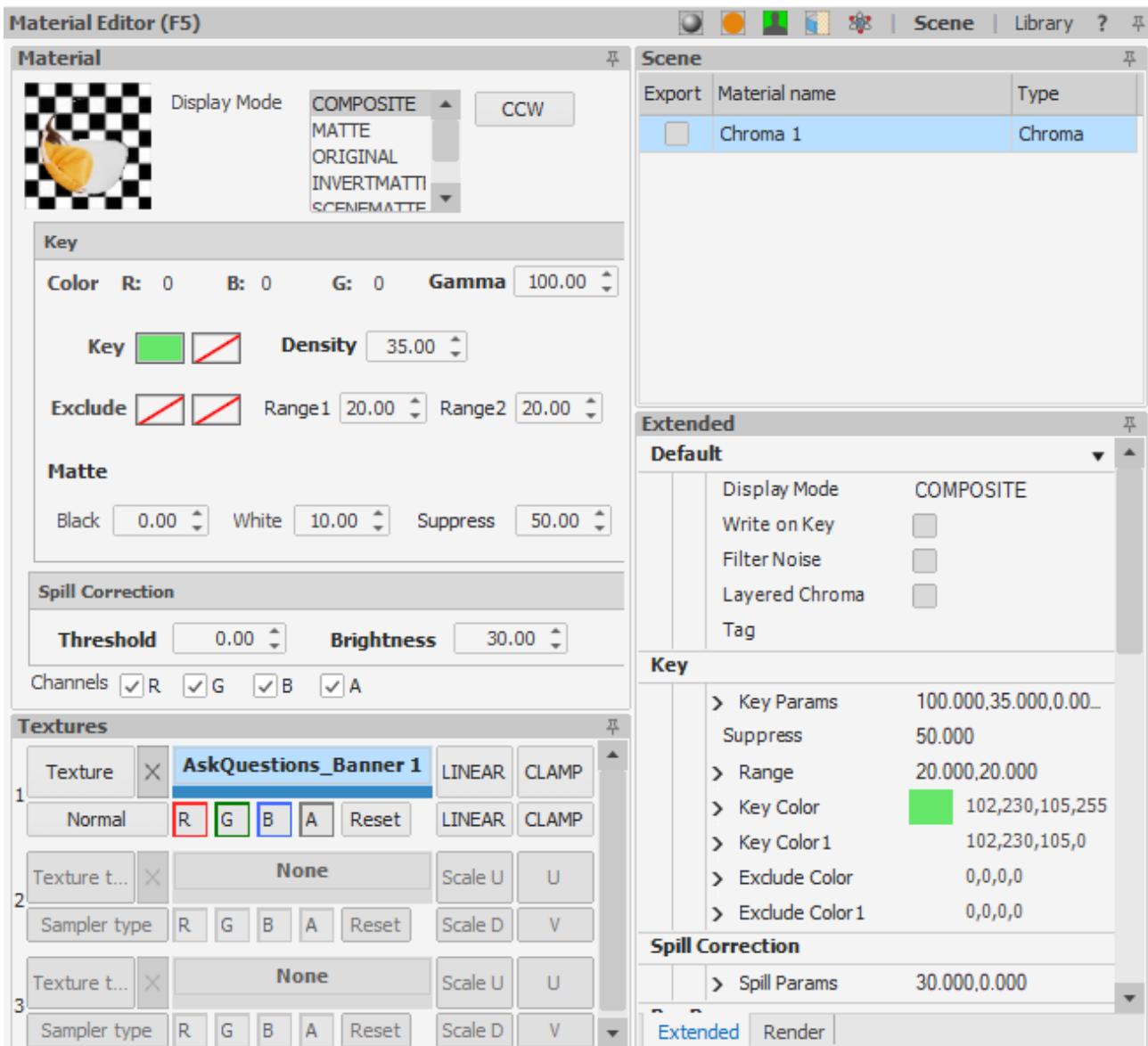
Quad Wireframe Parameters

- **Density:** Sets the density of wire. Change fill mode to quad wire-frame to see the changes.
- **Thickness:** Sets the size of wire, Change fill mode to quad wire-frame to see the changes.
- **Wire Blend:** Sets the blend between fill and wire-frame of the object, change fill mode to quad wire-frame to see the changes,

Light Parameters

- **Light:** Specifies the light for the material. A separate light (other than default light) can be specified to affect the material.

Chroma Material



Material Editor (F5)

It has 5 types of **Materials**

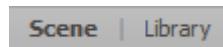


- Standard Material: This Material type is used on 3D shapes. The Standard material provides a reflective surface to an object. A surface displaying single color usually reflects many colors. Standard materials use a three-color model (Ambient, Diffuse and Specular) to show color blending and reflective surface. Apart from color scheme, the visual appearance of the surface is determined by the reflective properties, transparency properties and texture layers. Also, Standard Material shows an effect of light and different lights can be assigned for different Materials. It can be used for creating most of the real world materials like Wood, Plastic, Steel, etc.
- Uniform: This Material type is used with 2D shapes. It is a non-reflective material and has no effect of light. The Diffuse color component provides the Fill color to the material. Multiple

texture layers, Texture Types along with their blending options can be used with this material. Uniform material as four texture layers.

- Chroma: It provides real-time internal Chroma-keying. It allows blending between a live video texture source (shot on a blue/green background) and key out (make transparent) the background from this video texture in real-time. Chroma material contains two texture layers only. This helpful when the user wants to apply opacity map on the Chroma object.
- Matte: It is applied to the objects to make them invisible and block all objects behind it except the background. It has no color and texture layers. Matte Material can be used for selected objects using the Stencil-matte property of the object available in Parameter pool (F4).
- Particle: This Material type is used on particle object. By applying this material to a particle, you can make a particle change its color according to its life (as in frames), by defining the particle Start Color, Mid Color and End Color. Particle material can also use a grid based sprite and DDS volume texture to use the sprites as different particles of the particle object. The sprite texture can be divided using the Row and Columns properties in Extended Parameters -> Particle Texture Parameters.

The Material Pool window has two tabs:



- **Scene** – Provides an interface to define & modify materials.
- **Library** – Has as a collection of materials kept as references or for repeated use on the user's local system. It enables users to create/delete folder, view materials in Large Icon/Details, etc.

Channels - The Channels section allows users to manage the RGBA (Red, Green, Blue, Alpha) output channels for a material. Each channel can be individually enabled or disabled based on the specific design requirements.

For instance, in Virtual Set environments where objects are intended to appear behind a chroma key, the Alpha Channel is typically disabled to ensure proper compositing. If transparency is needed, users can simply enable the Alpha channel by checking the corresponding option.

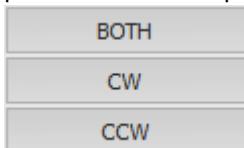
This feature offers precise control over how materials are rendered and composited within the scene.

1. Chroma Material



- Display Mode:
 - Composite: Shows the final Chroma output.
 - Matte: Shows the greyscale matte layer generated
 - Original: Shows the original texture map without Chroma
 - Invertmatte:
 - Scenematte:
 - Compositematte:

- Cullmode: Determines which side(s) of polygons are not rendered, optimizing rendering performance and preventing unnecessary calculations for unseen faces.



- BOTH: This term might refer to rendering or display modes where both front-facing and back-facing polygons or surfaces are visible and rendered. Typically, in many 3D engines or software, you can choose to render both sides of polygons (front and back) or only the front-facing side for optimization purposes.
- CCW: Abbreviation for "Counter ClockWise," often used in graphics to define the order of vertices that determine the orientation of a polygon face.
- CW: Stands for "Clockwise." In 3D graphics, it refers to the order in which vertices of a polygon are arranged to define the face of that polygon. When vertices are arranged in a clockwise order, it determines the front-facing side of the polygon.

Scene

Scene		
Export	Material name	Type
<input type="checkbox"/>	S_Cube 1	Standard
<input type="checkbox"/>	S_Cylinder 1	Standard
<input type="checkbox"/>	S_N_Gon 1	Standard
<input type="checkbox"/>	S_Ring 1	Standard
<input type="checkbox"/>	S_Sphere 1	Standard

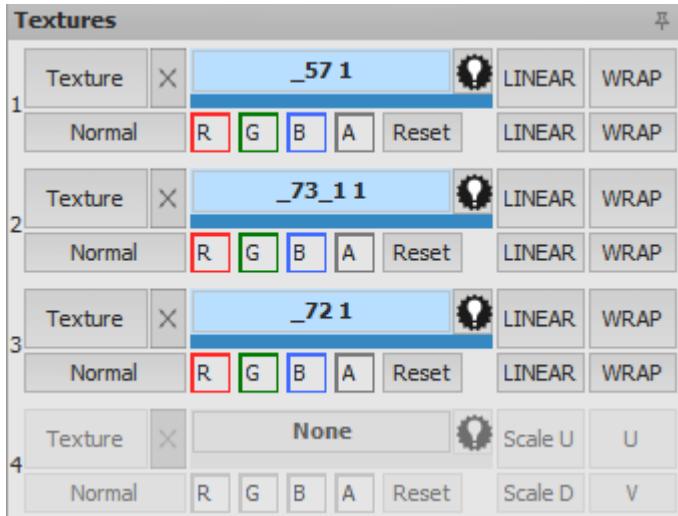
Export: When enabled, it allows selecting or applying the respective material from external applications.

Material name: Displays the name of the Materials

Type: Displays the type of Material.

User can Drag and Drop layers to change the layer order. Click on the Layer Number and Drag Up/ Down using mouse. Press Shift+Drag to Swap between the selected texture layers.

- Textures

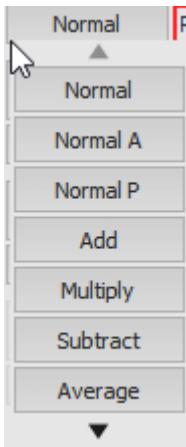


Texture Type defines the behavior of texture when mixed with material.



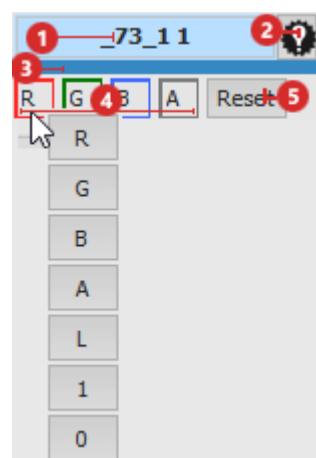
- Texture: The Texture option wraps a bitmap image onto an object's surface to create a visually realistic object. A bitmap consists of texture pixels or textures, where each texture provides the visual complexity which otherwise would have to be modeled. This advantage of lower polygon numbers helps reduce the rendering power while maintaining the frame-rate. X (X-Ray) option is available to enable X-Ray mode for texture when applied to a material of an object, the texture is scaled to the size of the viewport/render-area. The fill area of the object works as a window to reveal the texture.
- Opacity: Opacity is a kind of texture-map where the bitmap wrapped on the surface, is used to define the alpha (transparency) of the object surface. X option enables the selected texture to work in X-ray mode.
- Gradient: Gradient wipes the Luminance value of the selected channel.
- Mirror: The texture is mirrored and then tiled.

Normal - Filters



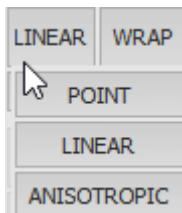
- Normal: The default mode, where no blending between texture maps occurs. Each texture pixel on the surface is painted as it appears in the original texture bitmap. Blending occurs between the base color and blend color based on the transparency of the texture pixels.
- Normal P: Works with Premultiplied Alpha, where the texture is blended with the layer behind it based on its alpha values. This mode assumes the colors in the texture have already been multiplied by the alpha, resulting in smoother blending, especially for semi-transparent textures.
- Normal A: In this mode, the alpha is multiplied separately, and colors are then blended based on their transparency. This method provides more control over how the colors interact with the layers behind, often leading to more precise edge blending.
- Add: This mode adds the base color to the blend color, resulting in a brighter color that moves towards white.
- Multiply: Multiplies the base color by the blend color, which results in a darker image. It is commonly used for shadow effects because multiplying any color by black results in black, while multiplying by white leaves the color unchanged.
- Subtract: Subtracts the blend color from the base color or vice versa, depending on which has a greater brightness value. This moves the result towards black.
- Average: Averages all the color components (RGBA) of the base and blend colors, resulting in a mid-tone between the two.
- Screen: Both the base and blend colors are inverted, multiplied by each other, and then the result is inverted again. Screening with black leaves the color unchanged, while screening with white results in white.
- Darken: Compares the base and blend colors and takes the darkest one as the output color.
- Lighten: Compares the base and blend colors and takes the lightest one as the output color.
- Exclusion: Produces a result where black and white return the same result, but colors in between become gray. This mode creates a "difference" effect but with lower contrast, and can be used to invert parts of an image.
- Overlay: A combination of multiply and screen modes, depending on the base color. If the base color is light, the blend color is screened. If the base color is dark, the blend color is multiplied.
- Hard Light: Uses the blend color as the base and either multiplies or screens the colors, depending on the blend color. If the blend color is lighter than 50% gray, the base is screened; if darker, it's multiplied.
- Soft Light: Similar to hard light but with a more subtle effect. It uses the blend color as the base and darkens or lightens the colors, adding a soft contrast to the image.

- Color Dodge: Brightens the base color to reflect the blend color by dividing the base by the inverse of the blend color. This creates a highlight effect.
- Color Burn: Darkens the base color to reflect the blend color by inverting, dividing, and inverting again. It creates a burn or shadow effect.
- Classic Color Dodge: Similar to Color Dodge but follows a more traditional color dodge algorithm, producing a smoother blend.
- Classic Color Burn: Similar to Color Burn but uses a traditional burn algorithm, giving a more controlled darkening effect.
- Linear Dodge: Similar to Color Dodge but uses linear interpolation, resulting in a smoother brightening effect.
- Echo: Adds shine or light to a texture. The output resembles soft light but with a more pronounced luminance effect.
- Glow: Similar to Echo but with the base and blend colors swapped. The result is closer to hard light, emphasizing highlights.
- Freeze: A variation of the Echo mode, where the base color is inverted, the blend color is inverted, and the result is inverted again, creating a cooling or frosted effect.
- Heat: Similar to Freeze but with the base and blend colors swapped, producing a warming effect rather than a cooling one.
- Interpolation: Similar to Average mode but with increased contrast between the base and blend colors.
- Stamp: The output is a blend of the base and blend colors, similar to Average, but works best with medium-toned colors, creating a stamped or embossed effect.
- Invert: Inverts the base color. Black becomes white, white becomes black, and other colors are converted to their complementary values.



- Texture Name: Displaying the name of the texture.
- Inherit Scene Light: Implies objects in a 3D scene adopting the lighting characteristics of the overall scene rather than having individualized lighting settings.
- Opacity Slider: Manage the transparency levels of the texture.
- RGBA
 1. R: Represents the Red color channel in digital graphics, defining the intensity of red in a color.
 2. G: Represents the Green color channel, determining the intensity of green in a color.
 3. B: Represents the Blue color channel, indicating the intensity of blue in a color.

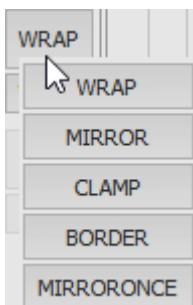
4. **A:** Denotes the Alpha channel, controlling transparency or opacity in addition to color.
 5. **L:** Represents the Luminance channel, often used to describe the brightness or lightness of a color.
 6. **1:** Often signifies the maximum value or full intensity in certain color channels, such as in some color models where 1 represents the highest value.
 7. **0:** Often represents the minimum value or absence of intensity in certain color channels, such as in some color models where 0 signifies no intensity or absence of color.
- **Reset:** Resetting the RGBA color values back to their default settings.
 - **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of he objects by specifying object. Tag as target.



Point: Color values of the closest adjacent pixel are taken to smoother the texture.

Linear: It is the default option & uses Bilinear Filtering in which a weighted average of four adjacent texture pixels is taken to create a single Texel.

Anisotropic: It takes into account the angle a primitive is rendered. Anisotropic compensates for distortion caused by the difference in angle between the texture polygon and the plane of the screen.



Wrap: Texture repeats n number of times in the U direction based on the value provided.

Mirror: The texture is mirrored and then tiled.

Clamp: Restricts the texture look up to the range 0 to 1. The unmapped area is mapped with the last pixel value.

Border: Generates a border for the texturemap.

Mirroronce: Similar to Mirror, the texture is mirrored only once and not tiled.

Extended

Displays extended set of parameters of the respective material.

Extended		
Default		
	Display Mode	INVERTMATTE
	Write on Key	<input type="checkbox"/>
	Filter Noise	<input type="checkbox"/>
	Layered Chroma	<input type="checkbox"/>
	Tag	
Key		
	➤ Key Params	100.000,35.000,0.00...
	Suppress	50.000
	➤ Range	20.000,20.000
	➤ Key Color	 102,230,105,255
	➤ Key Color1	102,230,105,0
	➤ Exclude Color	0,0,0,0
	➤ Exclude Color1	0,0,0,0
Spill Correction		
	➤ Spill Params	30.000,0.000
Pre Process		
	Preprocess	<input type="checkbox"/>
	➤ Pre Color Correct...	1.000,1.000,1.000
	➤ Hue	0,0,0,0
	Tolerance	0.000
	Hue Rotation	0.000
Post Process		
	Brightness	100.000
	Contrast	100.000
	Saturation	100.000
	Postprocess	<input type="checkbox"/>
	Gamma	100.000
Faux Shadow Parameters		
	Faux Shadow	<input type="checkbox"/>
	Object Visibility	<input checked="" type="checkbox"/>
	Faux Shadow Blur	0.000
	Blur Ratio	1.000
	Faux Shadow Offset	0.000
	Faux Shadow Angle	0.000
	➤ Faux Shadow Color	 0,0,0,64
	Shadow Depth	5.000
3D Shadow Parameters		
	Shadow Cast	<input type="checkbox"/>
Material Color Channels		
	Red	<input checked="" type="checkbox"/>
	Green	<input checked="" type="checkbox"/>
	Blue	<input checked="" type="checkbox"/>
	Alpha	<input checked="" type="checkbox"/>

Default

- **Display Mode:**
 - Composite: Shows the final Chroma output,
 - Matte: Shows the greyscale matte layer generated.
 - Original: Shows the original Texture map without Chroma
- Write on Key: Toggles graphic to be written on the Chroma area. This is useful for field chroma graphics like Soccer, Cricket etc,
- Filter Noise: Toggles the Noise Filtering
- Layered Chroma: Toggles merging of VSet Layer and Chroma layer in the Chroma Material
- **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of the objects by specifying Object.Tag as Target.

Key

- **Key Params:** Determines the algorithm used based on the current pixel's green channel value. If the green value is below the threshold, the Green Density Algorithm is applied. If the green value is equal to or above the threshold, the Key Color Algorithm is used.
- Suppress: Applies a bias to the final alpha value after the nonlinear color difference has been computed. This effectively reduces or suppresses the alpha to fine-tune transparency and prevent unwanted spill.
- Range: Defines the range of the exclude color
- **Key Color :** Sets the Ke Color 'Nith independent Red Green Blue controls.
- Key Color1 : Sets the Ke Color 'Nith independent Red Green Blue controls.
- Exclude Color : Sets the Ke Color 'Nith independent Red Green Blue controls.
- Exclude Color 1 : Sets the Ke Color 'Nith independent Red Green Blue controls.
-

Spill Correction

- **Spill Param:** Spill Params

Pre Process

- **Spill Param:** Spill Params

Post Process

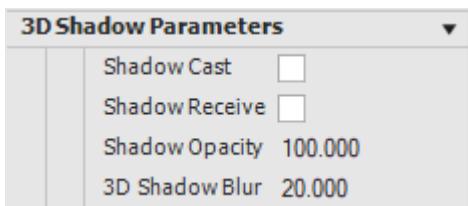
- **Spill Param:** Spill Params

Faux Shadow Parameters

- **Faux Shadow:** Toggles visibility of shadow of the material
- Object Visibility: Toggles the visibility of the objects on which the selected material is applied.
- Faux Shadow Blur: Sets the amount of blur of the faux shadow.
- **Blue Ratio:** Sets the scale of the blur of the faux shadow,
- **Faux Shadow Offset:** Sets the distance of shadow from the object shadow.
- **Faux Shadow Angel:** Sets the angle of refraction in respect to the object.
- **Faux Shadow Color:** Sets the color and alpha of the shadow.

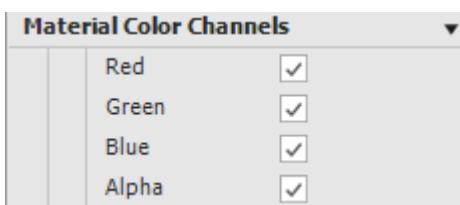
- **Shadow Depth:** Sets the shadow depth in respect to the object.

3D Shadow Parameters



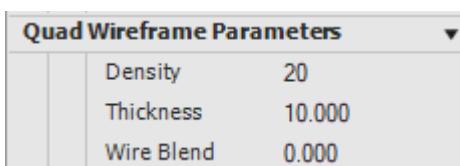
- **Shadow Cast:** Enables casting of shadow of the object to which the material is applied.
- **Shadow Receive:** Enables the object to receive 3D shadows on its surface to which the material is applied.
- **Shadow Opacity:** Sets a normalized value for opacity of the casted shadow. Works on the receiver object.
- **3D Shadow Blur:** Sets a normalized value for the amount of blur of the casted shadow. Works on the receiver object.

Material Color Channels



- **Red:** When ON, renders the Red channel of the Material.
- **Green:** When ON, renders the Green channel of the Material.
- **Blue:** When ON, renders the Blue channel of the Material.
- **Alpha:** When ON, renders the Alpha channel of the Material.

Quad Wireframe Parameters



- **Density:** Sets the density of wire. Change fill mode to quad wire-frame to see the changes.
- **Thickness:** Sets the size of wire. Change fill mode to quad wire-frame to see the changes.
- **Wire Blend:** Sets the blend between fill and wire-frame of the object, change fill mode to quad wire-frame to see the changes.

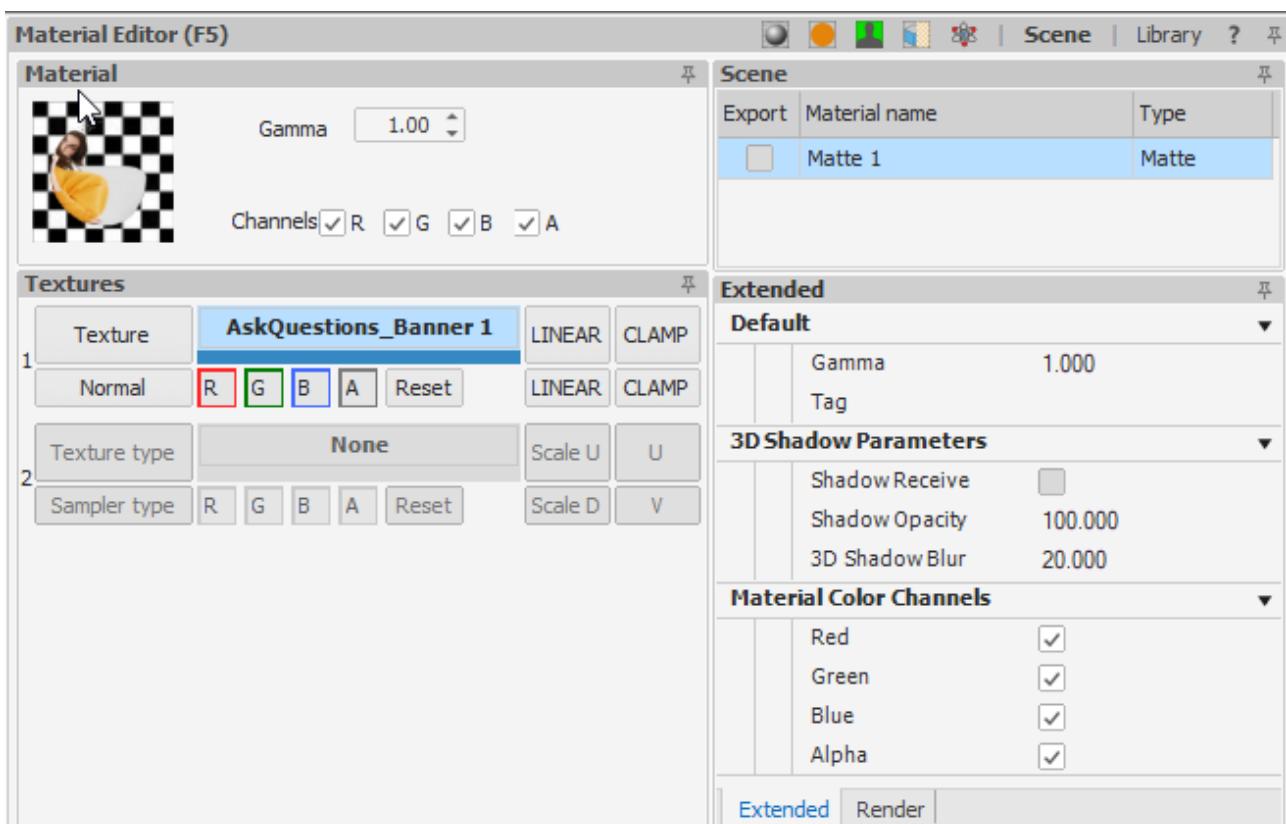
Light Parameters



- **Light Parameters:** Specifies the light for the material. A separate light (other than default light) can be specified to affect the material.

- Sampler Type: Provides various texture blending modes, which can be used to blend between textures applied in different layers of the material.
- Render States: Provides options like fill mode, cull mode, point size and object blending parameters. Object blending works on the entire object to which the respective material is applied.

Material Editor



Material Editor (F5)

It has 5 types of **Materials**

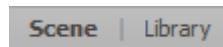


- Standard Material: This Material type is used on 3D shapes. The Standard material provides a reflective surface to an object. A surface displaying single color usually reflects many colors. Standard materials use a three-color model (Ambient, Diffuse and Specular) to show color blending and reflective surface. Apart from color scheme, the visual appearance of the surface is determined by the reflective properties, transparency properties and texture layers. Also, Standard Material shows an effect of light and different lights can be assigned for different Materials. It can be used for creating most of the real world materials like Wood, Plastic, Steel, etc.
- Uniform: This Material type is used with 2D shapes. It is a non-reflective material and has no effect of light. The Diffuse color component provides the Fill color to the material. Multiple

texture layers, Texture Types along with their blending options can be used with this material. Uniform material as four texture layers.

- Chroma: It provides real-time internal Chroma-keying. It allows blending between a live video texture source (shot on a blue/green background) and key out (make transparent) the background from this video texture in real-time. Chroma material contains two texture layers only. This helpful when the user wants to apply opacity map on the Chroma object.
- Matte: It is applied to the objects to make them invisible and block all objects behind it except the background. It has no color and texture layers. Matte Material can be used for selected objects using the Stencil-matte property of the object available in Parameter pool (F4).
- Particle: This Material type is used on particle object. By applying this material to a particle, you can make a particle change its color according to its life (as in frames), by defining the particle Start Color, Mid Color and End Color. Particle material can also use a grid based sprite and DDS volume texture to use the sprites as different particles of the particle object. The sprite texture can be divided using the Row and Columns properties in Extended Parameters -> Particle Texture Parameters.

The Material Pool window has two tabs:



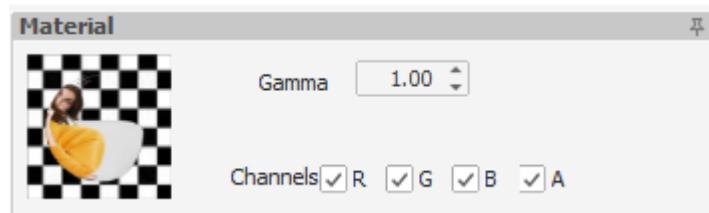
- **Scene** – Provides an interface to define & modify materials.
- **Library** – Has as a collection of materials kept as references or for repeated use on the user's local system. It enables users to create/delete folder, view materials in Large Icon/Details, etc.

Channels - The Channels section allows users to manage the RGBA (Red, Green, Blue, Alpha) output channels for a material. Each channel can be individually enabled or disabled based on the specific design requirements.

For instance, in Virtual Set environments where objects are intended to appear behind a chroma key, the Alpha Channel is typically disabled to ensure proper compositing. If transparency is needed, users can simply enable the Alpha channel by checking the corresponding option.

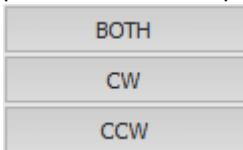
This feature offers precise control over how materials are rendered and composited within the scene.

1. Matte Material



- Gamma : A measure of brightness and contrast in digital imaging, affecting color accuracy and display.

- Cullmode: Determines which side(s) of polygons are not rendered, optimizing rendering performance and preventing unnecessary calculations for unseen faces.



- BOTH: This term might refer to rendering or display modes where both front-facing and back-facing polygons or surfaces are visible and rendered. Typically, in many 3D engines or software, you can choose to render both sides of polygons (front and back) or only the front-facing side for optimization purposes.
- CCW: Abbreviation for "Counter ClockWise," often used in graphics to define the order of vertices that determine the orientation of a polygon face.
- CW: Stands for "Clockwise." In 3D graphics, it refers to the order in which vertices of a polygon are arranged to define the face of that polygon. When vertices are arranged in a clockwise order, it determines the front-facing side of the polygon.
- Channels : The Channels section allows users to manage the RGBA (Red, Green, Blue, Alpha) output channels for a material. Each channel can be individually enabled or disabled based on the specific design requirements.

For instance, in Virtual Set environments where objects are intended to appear behind a chroma key, the Alpha Channel is typically disabled to ensure proper compositing. If transparency is needed, users can simply enable the Alpha channel by checking the corresponding option.

This feature offers precise control over how materials are rendered and composited within the scene.

Scene

Scene		
Export	Material name	Type
<input type="checkbox"/>	S_Cube 1	Standard
<input type="checkbox"/>	S_Cylinder 1	Standard
<input checked="" type="checkbox"/>	S_N_Gon 1	Standard
<input type="checkbox"/>	S_Ring 1	Standard
<input type="checkbox"/>	S_Sphere 1	Standard

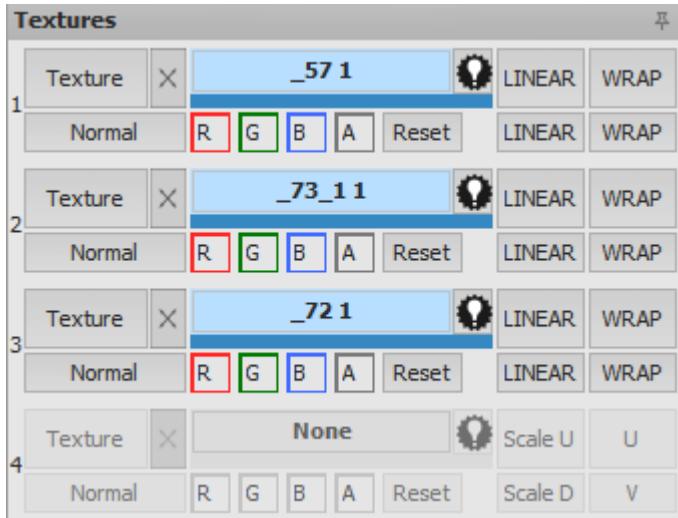
Export: When enabled, it allows selecting or applying the respective material from external applications.

Material name: Displays the name of the Materials

Type: Displays the type of Material.

User can Drag and Drop layers to change the layer order. Click on the Layer Number and Drag Up/ Down using mouse. Press Shift+Drag to Swap between the selected texture layers.

- Textures

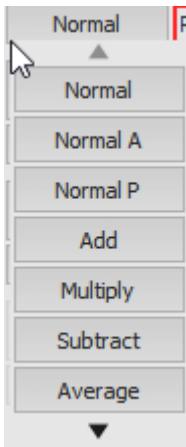


Texture Type defines the behavior of texture when mixed with material.



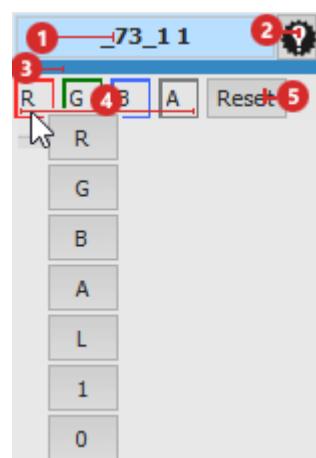
- Texture: The Texture option wraps a bitmap image onto an object's surface to create a visually realistic object. A bitmap consists of texture pixels or textures, where each texture provides the visual complexity which otherwise would have to be modeled. This advantage of lower polygon numbers helps reduce the rendering power while maintaining the frame-rate. X (X-Ray) option is available to enable X-Ray mode for texture when applied to a material of an object, the texture is scaled to the size of the viewport/render-area. The fill area of the object works as a window to reveal the texture.
- Opacity: Opacity is a kind of texture-map where the bitmap wrapped on the surface, is used to define the alpha (transparency) of the object surface. X option enables the selected texture to work in X-ray mode.
- Gradient: Gradient wipes the Luminance value of the selected channel.
- Mirror: The texture is mirrored and then tiled.

Normal - Filters



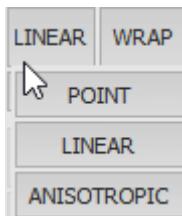
- Normal: The default mode, where no blending between texture maps occurs. Each texture pixel on the surface is painted as it appears in the original texture bitmap. Blending occurs between the base color and blend color based on the transparency of the texture pixels.
- Normal P: Works with Premultiplied Alpha, where the texture is blended with the layer behind it based on its alpha values. This mode assumes the colors in the texture have already been multiplied by the alpha, resulting in smoother blending, especially for semi-transparent textures.
- Normal A: In this mode, the alpha is multiplied separately, and colors are then blended based on their transparency. This method provides more control over how the colors interact with the layers behind, often leading to more precise edge blending.
- Add: This mode adds the base color to the blend color, resulting in a brighter color that moves towards white.
- Multiply: Multiplies the base color by the blend color, which results in a darker image. It is commonly used for shadow effects because multiplying any color by black results in black, while multiplying by white leaves the color unchanged.
- Subtract: Subtracts the blend color from the base color or vice versa, depending on which has a greater brightness value. This moves the result towards black.
- Average: Averages all the color components (RGBA) of the base and blend colors, resulting in a mid-tone between the two.
- Screen: Both the base and blend colors are inverted, multiplied by each other, and then the result is inverted again. Screening with black leaves the color unchanged, while screening with white results in white.
- Darken: Compares the base and blend colors and takes the darkest one as the output color.
- Lighten: Compares the base and blend colors and takes the lightest one as the output color.
- Exclusion: Produces a result where black and white return the same result, but colors in between become gray. This mode creates a "difference" effect but with lower contrast, and can be used to invert parts of an image.
- Overlay: A combination of multiply and screen modes, depending on the base color. If the base color is light, the blend color is screened. If the base color is dark, the blend color is multiplied.
- Hard Light: Uses the blend color as the base and either multiplies or screens the colors, depending on the blend color. If the blend color is lighter than 50% gray, the base is screened; if darker, it's multiplied.
- Soft Light: Similar to hard light but with a more subtle effect. It uses the blend color as the base and darkens or lightens the colors, adding a soft contrast to the image.

- Color Dodge: Brightens the base color to reflect the blend color by dividing the base by the inverse of the blend color. This creates a highlight effect.
- Color Burn: Darkens the base color to reflect the blend color by inverting, dividing, and inverting again. It creates a burn or shadow effect.
- Classic Color Dodge: Similar to Color Dodge but follows a more traditional color dodge algorithm, producing a smoother blend.
- Classic Color Burn: Similar to Color Burn but uses a traditional burn algorithm, giving a more controlled darkening effect.
- Linear Dodge: Similar to Color Dodge but uses linear interpolation, resulting in a smoother brightening effect.
- Echo: Adds shine or light to a texture. The output resembles soft light but with a more pronounced luminance effect.
- Glow: Similar to Echo but with the base and blend colors swapped. The result is closer to hard light, emphasizing highlights.
- Freeze: A variation of the Echo mode, where the base color is inverted, the blend color is inverted, and the result is inverted again, creating a cooling or frosted effect.
- Heat: Similar to Freeze but with the base and blend colors swapped, producing a warming effect rather than a cooling one.
- Interpolation: Similar to Average mode but with increased contrast between the base and blend colors.
- Stamp: The output is a blend of the base and blend colors, similar to Average, but works best with medium-toned colors, creating a stamped or embossed effect.
- Invert: Inverts the base color. Black becomes white, white becomes black, and other colors are converted to their complementary values.



- Texture Name: Displaying the name of the texture.
- Inherit Scene Light: Implies objects in a 3D scene adopting the lighting characteristics of the overall scene rather than having individualized lighting settings.
- Opacity Slider: Manage the transparency levels of the texture.
- RGBA
 1. R: Represents the Red color channel in digital graphics, defining the intensity of red in a color.
 2. G: Represents the Green color channel, determining the intensity of green in a color.
 3. B: Represents the Blue color channel, indicating the intensity of blue in a color.

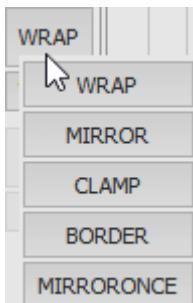
4. **A:** Denotes the Alpha channel, controlling transparency or opacity in addition to color.
 5. **L:** Represents the Luminance channel, often used to describe the brightness or lightness of a color.
 6. **1:** Often signifies the maximum value or full intensity in certain color channels, such as in some color models where 1 represents the highest value.
 7. **0:** Often represents the minimum value or absence of intensity in certain color channels, such as in some color models where 0 signifies no intensity or absence of color.
- **Reset:** Resetting the RGBA color values back to their default settings.
 - **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of he objects by specifying object. Tag as target.



Point: Color values of the closest adjacent pixel are taken to smoother the texture.

Linear: It is the default option & uses Bilinear Filtering in which a weighted average of four adjacent texture pixels is taken to create a single Texel.

Anisotropic: It takes into account the angle a primitive is rendered. Anisotropic compensates for distortion caused by the difference in angle between the texture polygon and the plane of the screen.



Wrap: Texture repeats n number of times in the U direction based on the value provided.

Mirror: The texture is mirrored and then tiled.

Clamp: Restricts the texture look up to the range 0 to 1. The unmapped area is mapped with the last pixel value.

Border: Generates a border for the texturemap.

Mirroronce: Similar to Mirror, the texture is mirrored only once and not tiled.

Extended

Displays extended set of parameters of the respective material.

Default		
	Shine	0.500
	Edge	0.800
	Incident	0.900
	Fresnel	1.000
	Refractivity Index	1.030
	Shuffle Repeat	0
	Texture Cols	1
	Texture Rows	1
	Gamma	1.000
	Cubic Node	None
	Use Ambient Map	<input checked="" type="checkbox"/>
	Tag	

Default

- **Shine:** Defines the strength/intensity of the specular color on the surface of the object.
- **Edge:** Sets the intensity of refection from the front of the object,
- Incident: Sets the intensity of refection from the front of the object.
- Fresnel: Sets the blend between the reflection Min and Max.
- Refractivity Index: Specifies the refractivity Index of the texture applied in the refraction texture mode.
- Roughness: Specifies the roughness of the material when a roughness mode is selected for a texture in the texture layer.
- Shuffle Repeat: Specifies the phase value for he texture cell animation. If it is 0, It would not repeat, when 1 it starts to shuffle across the texture cells.
- Texture Cols: Specifies the vertical division in the texture of the material. This can be used to animate the material as a sprite.
- Texture Row: Specifies the horizontal division in the texture of the material. This can be used to animate the material as a sprite.
- **Gamma:** Sets Gamma properties to Increase or decrease the color intensity of the material.
- Cubic Node: Reference node for local cubic map.
- Use Ambient Map: Applies Ambient light according to the Environment map.

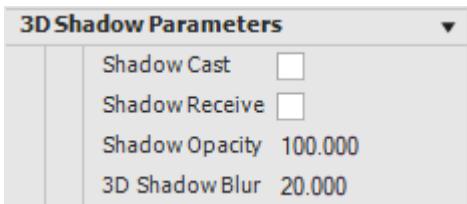
Faux Shadow Parameters

Faux Shadow Parameters	
	Faux Shadow <input type="checkbox"/>
	Object Visibility <input checked="" type="checkbox"/>
	Faux Shadow B... 0.000
	Blur Ratio 1.000
	Faux Shadow O... 0
	Faux Shadow A... 0
➤	Faux Shado... 0,0,0,200
	Shadow Depth 5.000

- **Faux Shadow:** Toggles visibility of shadow of the material
- Object Visibility: Toggles the visibility of the objects on which the selected material is applied.

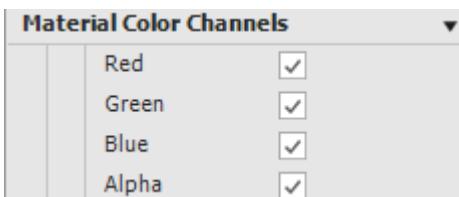
- **Faux Shadow Blur:** Sets the amount of blur of the faux shadow.
- **Blue Ratio:** Sets the scale of the blur of the faux shadow,
- **Faux Shadow Offset:** Sets the distance of shadow from the object shadow.
- **Faux Shadow Angel:** Sets the angle of refection in respect to the object.
- **Faux Shadow Color:** Sets the color and alpha of the shadow.
- **Shadow Depth:** Sets the shadow depth in respect to the object.

3D Shadow Parameters



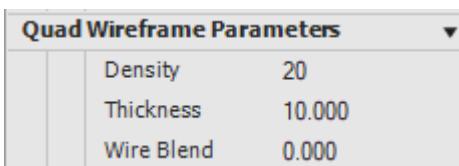
- **Shadow Cast:** Enables casting of shadow of the object to which the material is applied.
- **Shadow Receive:** Enables the object to receive 3D shadows on its surface to which the material applied.
- **Shadow Opacity:** Sets a normalized value for opacity of the casted shadow. Works on the receiver object.
- **3D Shadow Blue:** Sets a normalized value for the amount of blur of the casted shadow. Works on the receiver object.

Material Color Channels



- **Red:** When ON, renders the Red channel of the Material.
- **Green:** When ON, renders the Green channel of the Material.
- **Blue:** When ON, renders the Blue channel of the Material.
- **Alpha:** When ON, renders the Alpha channel of the Material.

Quad Wireframe Parameters



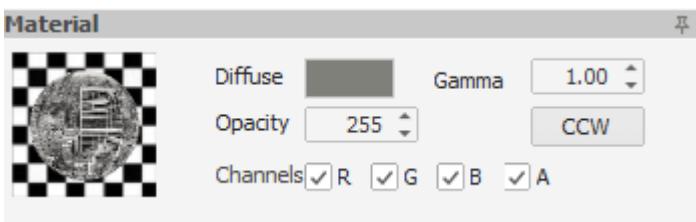
- **Density:** Sets the density of wire. Change fill mode to quad wire-frame to see the changes.
- **Thickness:** Sets the size of wire, Change fill mode to quad wire-frame to see the changes.
- **Wire Blend:** Sets the blend between fill and wire-frame of the object, change fill mode to quad wire-frame to see the changes,

Light Parameters

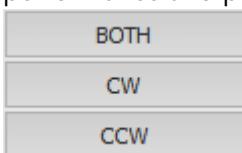


- Light Parameters: Specifies the light for the material. A separate light (other than default light) can be specified to affect the material.
- Sampler Type: Provides various texture blending modes, which can be used to blend between textures applied in different layers of the material.
- Render States: Provides options like fill mode, cull mode, point size and object blending parameters. Object blending works on the entire object to which the respective material is applied.

2. Material - Uniform

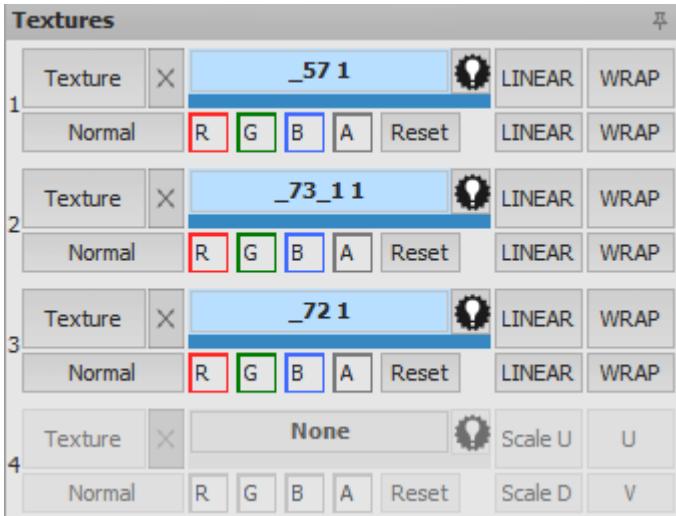


- Diffuse : The inherent surface color or texture of an object that evenly scatters light, determining its appearance under direct lighting.
- Gamma : A measure of brightness and contrast in digital imaging, affecting color accuracy and display.
- Opacity : The degree to which the color of an object or material is transparent or visible. Higher opacity means less transparency.
- Cullmode: Determines which side(s) of polygons are not rendered, optimizing rendering performance and preventing unnecessary calculations for unseen faces.



- BOTH: This term might refer to rendering or display modes where both front-facing and back-facing polygons or surfaces are visible and rendered. Typically, in many 3D engines or software, you can choose to render both sides of polygons (front and back) or only the front-facing side for optimization purposes.
- CCW: Abbreviation for "Counter ClockWise," often used in graphics to define the order of vertices that determine the orientation of a polygon face.
- CW: Stands for "Clockwise." In 3D graphics, it refers to the order in which vertices of a polygon are arranged to define the face of that polygon. When vertices are arranged in a clockwise order, it determines the front-facing side of the polygon.

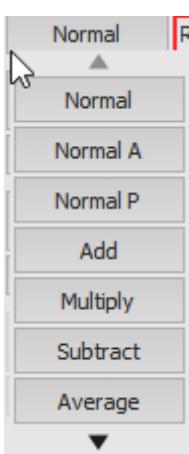
- Textures



Texture Type defines the behavior of texture when mixed with material.

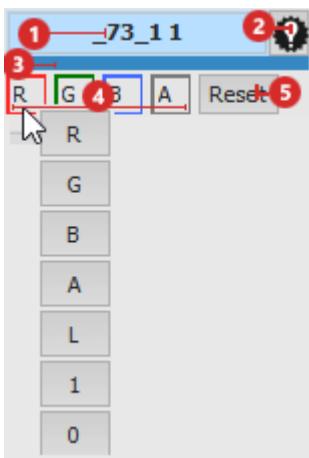


- Texture: The Texture option wraps a bitmap image onto an object's surface to create a visually realistic object. A bitmap consists of texture pixels or textures , where each texture provides the visual complexity which otherwise would have to be modeled. This advantage of lower polygon numbers helps reduce the rendering power while maintaining the frame-rate. X (X-Ray) option is available to enable X-Ray mode for texture when applied to a material of an object, the texture is scaled to the size of the viewport/render-area. The fill area of the object works as a window to reveal the texture.
- Opacity: Opacity is a kind of texture-map where the bitmap wrapped on the surface, is used to define the alpha (transparency) of the object surface. X option enables the selected texture to work in X-ray mode.
- Gradient: Gradient wipes the Luminance value of the selected channel.
- Mirror: The texture is mirrored and then tiled.
- Normal - Filters

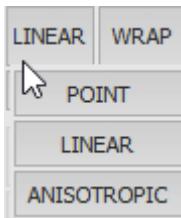


- Normal: The default mode, where no blending between texture maps occurs. Each texture pixel on the surface is painted as it appears in the original texture bitmap. Blending occurs between the base color and blend color based on the transparency of the texture pixels.
- Normal P: Works with Premultiplied Alpha, where the texture is blended with the layer behind it based on its alpha values. This mode assumes the colors in the texture have already been multiplied by the alpha, resulting in smoother blending, especially for semi-transparent textures.
- Normal A: In this mode, the alpha is multiplied separately, and colors are then blended based on their transparency. This method provides more control over how the colors interact with the layers behind, often leading to more precise edge blending.
- Add: This mode adds the base color to the blend color, resulting in a brighter color that moves towards white.
- Multiply: Multiplies the base color by the blend color, which results in a darker image. It is commonly used for shadow effects because multiplying any color by black results in black, while multiplying by white leaves the color unchanged.
- Subtract: Subtracts the blend color from the base color or vice versa, depending on which has a greater brightness value. This moves the result towards black.
- Average: Averages all the color components (RGBA) of the base and blend colors, resulting in a mid-tone between the two.
- Screen: Both the base and blend colors are inverted, multiplied by each other, and then the result is inverted again. Screening with black leaves the color unchanged, while screening with white results in white.
- Darken: Compares the base and blend colors and takes the darkest one as the output color.
- Lighten: Compares the base and blend colors and takes the lightest one as the output color.
- Exclusion: Produces a result where black and white return the same result, but colors in between become gray. This mode creates a "difference" effect but with lower contrast, and can be used to invert parts of an image.
- Overlay: A combination of multiply and screen modes, depending on the base color. If the base color is light, the blend color is screened. If the base color is dark, the blend color is multiplied.
- Hard Light: Uses the blend color as the base and either multiplies or screens the colors, depending on the blend color. If the blend color is lighter than 50% gray, the base is screened; if darker, it's multiplied.
- Soft Light: Similar to hard light but with a more subtle effect. It uses the blend color as the base and darkens or lightens the colors, adding a soft contrast to the image.
- Color Dodge: Brightens the base color to reflect the blend color by dividing the base by the inverse of the blend color. This creates a highlight effect.
- Color Burn: Darkens the base color to reflect the blend color by inverting, dividing, and inverting again. It creates a burn or shadow effect.
- Classic Color Dodge: Similar to Color Dodge but follows a more traditional color dodge algorithm, producing a smoother blend.
- Classic Color Burn: Similar to Color Burn but uses a traditional burn algorithm, giving a more controlled darkening effect.
- Linear Dodge: Similar to Color Dodge but uses linear interpolation, resulting in a smoother brightening effect.
- Echo: Adds shine or light to a texture. The output resembles soft light but with a more pronounced luminance effect.

- Glow: Similar to Echo but with the base and blend colors swapped. The result is closer to hard light, emphasizing highlights.
- Freeze: A variation of the Echo mode, where the base color is inverted, the blend color is inverted, and the result is inverted again, creating a cooling or frosted effect.
- Heat: Similar to Freeze but with the base and blend colors swapped, producing a warming effect rather than a cooling one.
- Interpolation: Similar to Average mode but with increased contrast between the base and blend colors.
- Stamp: The output is a blend of the base and blend colors, similar to Average, but works best with medium-toned colors, creating a stamped or embossed effect.
- Invert: Inverts the base color. Black becomes white, white becomes black, and other colors are converted to their complementary values.
-



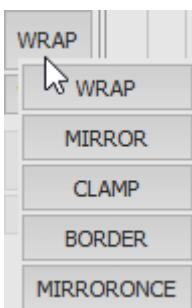
- Texture Name: Displaying the name of the texture.
- Inherit Scene Light: Implies objects in a 3D scene adopting the lighting characteristics of the overall scene rather than having individualized lighting settings.
- Opacity Slider: Manage the transparency levels of the texture.
- RGBA
 1. R: Represents the Red color channel in digital graphics, defining the intensity of red in a color.
 2. G: Represents the Green color channel, determining the intensity of green in a color.
 3. B: Represents the Blue color channel, indicating the intensity of blue in a color.
 4. A: Denotes the Alpha channel, controlling transparency or opacity in addition to color.
 5. **L:** Represents the Luminance channel, often used to describe the brightness or lightness of a color.
 6. **1:** Often signifies the maximum value or full intensity in certain color channels, such as in some color models where 1 represents the highest value.
 7. **0:** Often represents the minimum value or absence of intensity in certain color channels, such as in some color models where 0 signifies no intensity or absence of color.
- **Reset:** Resetting the RGBA color values back to their default settings.
- **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of he objects by specifying object. Tag as target.



Point: Color values of the closest adjacent pixel are taken to smoother the texture.

Linear: It is the default option & uses Bilinear Filtering in which a weighted average of four adjacent texture pixels is taken to create a single Texel.

Anisotropic: It takes into account the angle a primitive is rendered. Anisotropic compensates for distortion caused by the difference in angle between the texture polygon and the plane of the screen.



Wrap: Texture repeats n number of times in the U direction based on the value provided.

Mirror: The texture is mirrored and then tiled.

Clamp: Restricts the texture look up to the range 0 to 1. The unmapped area is mapped with the last pixel value.

Border: Generates a border for the texturemap.

Mirroronce: Similar to Mirror, the texture is mirrored only once and not tiled.

Extended

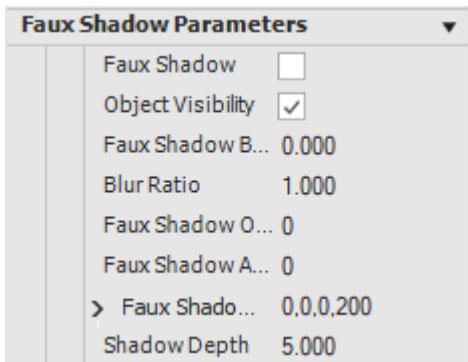
Displays extended set of parameters of the respective material.

Default		
	Shine	0.500
	Edge	0.800
	Incident	0.900
	Fresnel	1.000
	Refractivity Index	1.030
	Shuffle Repeat	0
	Texture Cols	1
	Texture Rows	1
	Gamma	1.000
	Cubic Node	None
	Use Ambient Map	<input checked="" type="checkbox"/>
	Tag	

Default

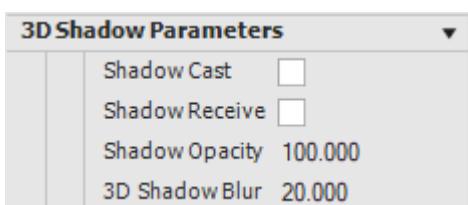
- **Shine:** Defines the strength/intensity of the specular color on the surface of the object.
- **Edge:** Sets the intensity of refection from the front of the object,
- Incident: Sets the intensity of refection from the front of the object.
- Fresnel: Sets the blend between the reflection Min and Max.
- Refractivity Index: Specifies the refractivity Index of the texture applied in the refraction texture mode.
- Roughness: Specifies the roughness of the material when a roughness mode is selected for a texture in the texture layer.
- Shuffle Repeat: Specifies the phase value for he texture cell animation. If it is 0, It would not repeat, when 1 it starts to shuffle across the texture cells.
- Texture Cols: Specifies the vertical division in the texture of the material. This can be used to animate the material as a sprite.
- Texture Row: Specifies the horizontal division in the texture of the material. This can be used to animate the material as a sprite.
- **Gamma:** Sets Gamma properties to Increase or decrease the color intensity of the material.
- Cubic Node: Reference node for local cubic map.
- Use Ambient Map: Applies Ambient light according to the Environment map.

Faux Shadow Parameters



- **Faux Shadow:** Toggles visibility of shadow of the material
- Object Visibility: Toggles the visibility of the objects on which the selected material is applied.
- Faux Shadow Blur: Sets the amount of blur of the faux shadow.
- **Blue Ratio:** Sets the scale of the blur of the faux shadow,
- **Faux Shadow Offset:** Sets the distance of shadow from the object shadow.
- **Faux Shadow Angel:** Sets the angle of refection in respect to the object.
- **Faux Shadow Color:** Sets the color and alpha of the shadow.
- **Shadow Depth:** Sets he Shadow depth in respect to the object.

3D Shadow Parameters



- **Shadow Cast:** Enables casting of shadow of the object to which the material is applied.
- **Shadow Receive:** Enables the object to receive 3D shadows on its surface to which the material applied.
- **Shadow Opacity:** Sets a normalized value for opacity of the casted shadow. Works on the receiver object.
- **3D Shadow Blue:** Sets a normalized value for the amount of blur of the casted shadow. Works on the receiver object.

Material Color Channels

Material Color Channels		
	Red	<input checked="" type="checkbox"/>
	Green	<input checked="" type="checkbox"/>
	Blue	<input checked="" type="checkbox"/>
	Alpha	<input checked="" type="checkbox"/>

- **Red:** When ON, renders the Red channel of the Material.
- **Green:** When ON, renders the Green channel of the Material.
- **Blue:** When ON, renders the Blue channel of the Material.
- **Alpha:** When ON, renders the Alpha channel of the Material.

Quad Wireframe Parameters

Quad Wireframe Parameters		
	Density	20
	Thickness	10.000
	Wire Blend	0.000

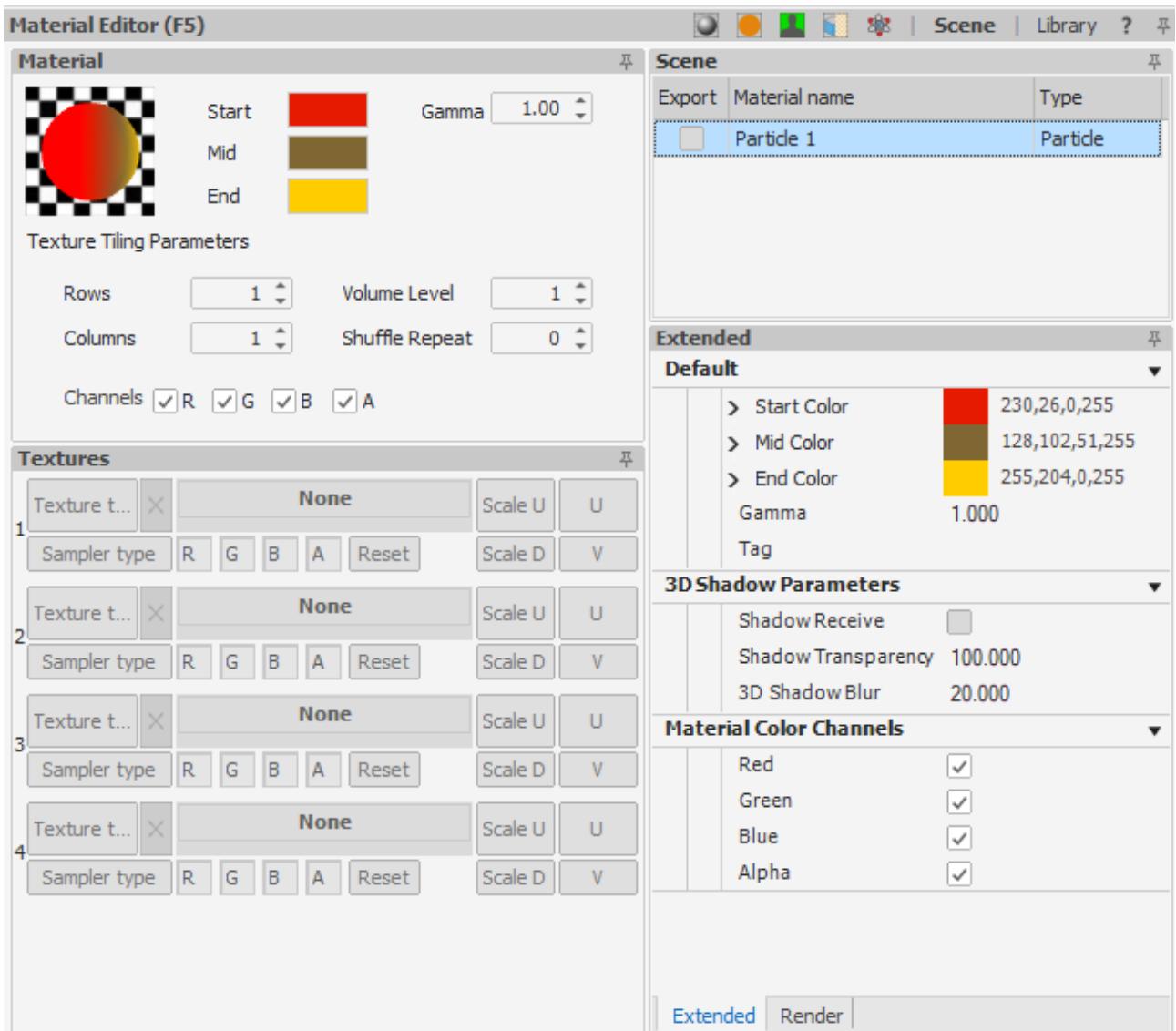
- **Density:** Sets the density of wire. Change fill mode to quad wire-frame to see the changes.
- **Thickness:** Sets the size of wire, Change fill mode to quad wire-frame to see the changes.
- **Wire Blend:** Sets the blend between fill and wire-frame of the object, change fill mode to quad wire-frame to see the changes,

Light Parameters

Light Parameters		
	Light	Directional Light 0

- **Light Parameters:** Specifies the light for the material. A separate light(other than default light) can be specified to affect the material.
- **Sampler Type:** Provides various texture blending modes, which can be used to blend between textures applied in different layers of the material.
- **Render States:** Provides options like fill mode, cull mode, point size and object blending parameters. Object blending works on the entire object to which the respective material is applied.

Particle Material



Material Editor (F5)

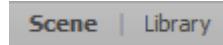
It has 5 types of **Materials**



- Standard Material: This Material type is used on 3D shapes. The Standard material provides a reflective surface to an object. A surface displaying single color usually reflects many colors. Standard materials use a three-color model (Ambient, Diffuse and Specular) to show color blending and reflective surface. Apart from color scheme, the visual appearance of the surface is determined by the reflective properties, transparency properties and texture layers. Also, Standard Material shows an effect of light and different lights can be assigned for different Materials. It can be used for creating most of the real world materials like Wood, Plastic, Steel, etc.
- Uniform: This Material type is used with 2D shapes. It is a non-reflective material and has no effect of light. The Diffuse color component provides the Fill color to the material. Multiple texture layers, Texture Types along with their blending options can be used with this material. Uniform material as four texture layers.

- Chroma: It provides real-time internal Chroma-keying. It allows blending between a live video texture source (shot on a blue/green background) and key out (make transparent) the background from this video texture in real-time. Chroma material contains two texture layers only. This helpful when the user wants to apply opacity map on the Chroma object.
- Matte: It is applied to the objects to make them invisible and block all objects behind it except the background. It has no color and texture layers. Matte Material can be used for selected objects using the Stencil-matte property of the object available in Parameter pool (F4).
- Particle: This Material type is used on particle object. By applying this material to a particle, you can make a particle change its color according to its life (as in frames), by defining the particle Start Color, Mid Color and End Color. Particle material can also use a grid based sprite and DDS volume texture to use the sprites as different particles of the particle object. The sprite texture can be divided using the Row and Columns properties in Extended Parameters -> Particle Texture Parameters.

The Material Pool window has two tabs:



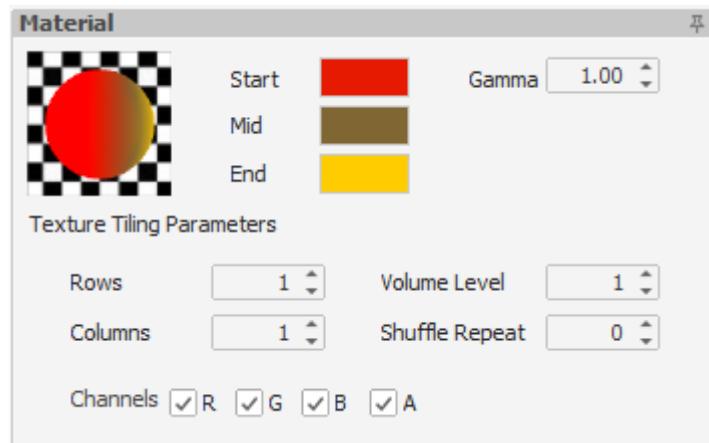
- **Scene** – Provides an interface to define & modify materials.
- **Library** – Has as a collection of materials kept as references or for repeated use on the user's local system. It enables users to create/delete folder, view materials in Large Icon/Details, etc.

Channels - The Channels section allows users to manage the RGBA (Red, Green, Blue, Alpha) output channels for a material. Each channel can be individually enabled or disabled based on the specific design requirements.

For instance, in Virtual Set environments where objects are intended to appear behind a chroma key, the Alpha Channel is typically disabled to ensure proper compositing. If transparency is needed, users can simply enable the Alpha channel by checking the corresponding option.

This feature offers precise control over how materials are rendered and composited within the scene.

1. Particle Material



- Start:
- Gamma : A measure of brightness and contrast in digital imaging, affecting color accuracy and display.

- Mid:
- End:
- Texture Tiling Parameters:
 - Rows:
 - Volume Level:
 - Columns:
 - Shuffle Repeat:
- Channels : The Channels section allows users to manage the RGBA (Red, Green, Blue, Alpha) output channels for a material. Each channel can be individually enabled or disabled based on the specific design requirements.
 For instance, in Virtual Set environments where objects are intended to appear behind a chroma key, the Alpha Channel is typically disabled to ensure proper compositing. If transparency is needed, users can simply enable the Alpha channel by checking the corresponding option. This feature offers precise control over how materials are rendered and composited within the scene.

Scene

Scene		
Export	Material name	Type
<input type="checkbox"/>	S_Cube 1	Standard
<input type="checkbox"/>	S_Cylinder 1	Standard
<input checked="" type="checkbox"/>	S_N_Gon 1	Standard
<input type="checkbox"/>	S_Ring 1	Standard
<input type="checkbox"/>	S_Sphere 1	Standard

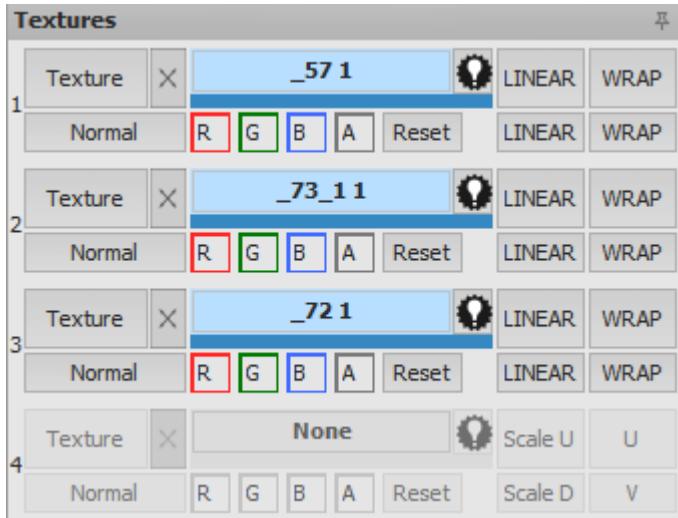
Export: When enabled, it allows selecting or applying the respective material from external applications.

Material name: Displays the name of the Materials

Type: Displays the type of Material.

User can Drag and Drop layers to change the layer order. Click on the Layer Number and Drag Up/ Down using mouse. Press Shift+Drag to Swap between the selected texture layers.

- Textures

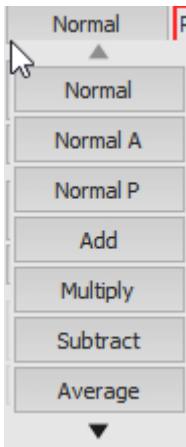


Texture Type defines the behavior of texture when mixed with material.



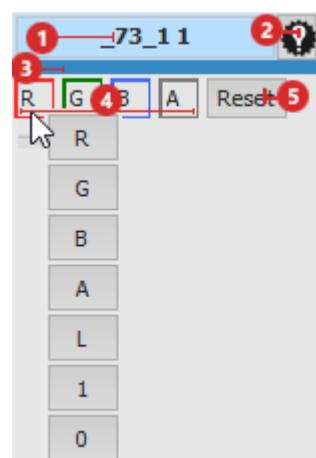
- Texture: The Texture option wraps a bitmap image onto an object's surface to create a visually realistic object. A bitmap consists of texture pixels or textures, where each texture provides the visual complexity which otherwise would have to be modeled. This advantage of lower polygon numbers helps reduce the rendering power while maintaining the frame-rate. X (X-Ray) option is available to enable X-Ray mode for texture when applied to a material of an object, the texture is scaled to the size of the viewport/render-area. The fill area of the object works as a window to reveal the texture.
- Opacity: Opacity is a kind of texture-map where the bitmap wrapped on the surface, is used to define the alpha (transparency) of the object surface. X option enables the selected texture to work in X-ray mode.
- Gradient: Gradient wipes the Luminance value of the selected channel.
- Mirror: The texture is mirrored and then tiled.

Normal - Filters



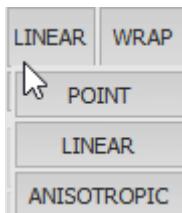
- Normal: The default mode, where no blending between texture maps occurs. Each texture pixel on the surface is painted as it appears in the original texture bitmap. Blending occurs between the base color and blend color based on the transparency of the texture pixels.
- Normal P: Works with Premultiplied Alpha, where the texture is blended with the layer behind it based on its alpha values. This mode assumes the colors in the texture have already been multiplied by the alpha, resulting in smoother blending, especially for semi-transparent textures.
- Normal A: In this mode, the alpha is multiplied separately, and colors are then blended based on their transparency. This method provides more control over how the colors interact with the layers behind, often leading to more precise edge blending.
- Add: This mode adds the base color to the blend color, resulting in a brighter color that moves towards white.
- Multiply: Multiplies the base color by the blend color, which results in a darker image. It is commonly used for shadow effects because multiplying any color by black results in black, while multiplying by white leaves the color unchanged.
- Subtract: Subtracts the blend color from the base color or vice versa, depending on which has a greater brightness value. This moves the result towards black.
- Average: Averages all the color components (RGBA) of the base and blend colors, resulting in a mid-tone between the two.
- Screen: Both the base and blend colors are inverted, multiplied by each other, and then the result is inverted again. Screening with black leaves the color unchanged, while screening with white results in white.
- Darken: Compares the base and blend colors and takes the darkest one as the output color.
- Lighten: Compares the base and blend colors and takes the lightest one as the output color.
- Exclusion: Produces a result where black and white return the same result, but colors in between become gray. This mode creates a "difference" effect but with lower contrast, and can be used to invert parts of an image.
- Overlay: A combination of multiply and screen modes, depending on the base color. If the base color is light, the blend color is screened. If the base color is dark, the blend color is multiplied.
- Hard Light: Uses the blend color as the base and either multiplies or screens the colors, depending on the blend color. If the blend color is lighter than 50% gray, the base is screened; if darker, it's multiplied.
- Soft Light: Similar to hard light but with a more subtle effect. It uses the blend color as the base and darkens or lightens the colors, adding a soft contrast to the image.

- Color Dodge: Brightens the base color to reflect the blend color by dividing the base by the inverse of the blend color. This creates a highlight effect.
- Color Burn: Darkens the base color to reflect the blend color by inverting, dividing, and inverting again. It creates a burn or shadow effect.
- Classic Color Dodge: Similar to Color Dodge but follows a more traditional color dodge algorithm, producing a smoother blend.
- Classic Color Burn: Similar to Color Burn but uses a traditional burn algorithm, giving a more controlled darkening effect.
- Linear Dodge: Similar to Color Dodge but uses linear interpolation, resulting in a smoother brightening effect.
- Echo: Adds shine or light to a texture. The output resembles soft light but with a more pronounced luminance effect.
- Glow: Similar to Echo but with the base and blend colors swapped. The result is closer to hard light, emphasizing highlights.
- Freeze: A variation of the Echo mode, where the base color is inverted, the blend color is inverted, and the result is inverted again, creating a cooling or frosted effect.
- Heat: Similar to Freeze but with the base and blend colors swapped, producing a warming effect rather than a cooling one.
- Interpolation: Similar to Average mode but with increased contrast between the base and blend colors.
- Stamp: The output is a blend of the base and blend colors, similar to Average, but works best with medium-toned colors, creating a stamped or embossed effect.
- Invert: Inverts the base color. Black becomes white, white becomes black, and other colors are converted to their complementary values.



- Texture Name: Displaying the name of the texture.
- Inherit Scene Light: Implies objects in a 3D scene adopting the lighting characteristics of the overall scene rather than having individualized lighting settings.
- Opacity Slider: Manage the transparency levels of the texture.
- RGBA
 1. R: Represents the Red color channel in digital graphics, defining the intensity of red in a color.
 2. G: Represents the Green color channel, determining the intensity of green in a color.
 3. B: Represents the Blue color channel, indicating the intensity of blue in a color.

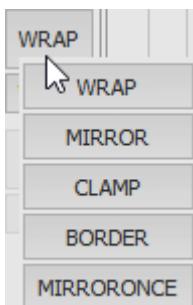
4. **A:** Denotes the Alpha channel, controlling transparency or opacity in addition to color.
 5. **L:** Represents the Luminance channel, often used to describe the brightness or lightness of a color.
 6. **1:** Often signifies the maximum value or full intensity in certain color channels, such as in some color models where 1 represents the highest value.
 7. **0:** Often represents the minimum value or absence of intensity in certain color channels, such as in some color models where 0 signifies no intensity or absence of color.
- **Reset:** Resetting the RGBA color values back to their default settings.
 - **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of he objects by specifying object. Tag as target.



Point: Color values of the closest adjacent pixel are taken to smoother the texture.

Linear: It is the default option & uses Bilinear Filtering in which a weighted average of four adjacent texture pixels is taken to create a single Texel.

Anisotropic: It takes into account the angle a primitive is rendered. Anisotropic compensates for distortion caused by the difference in angle between the texture polygon and the plane of the screen.



Wrap: Texture repeats n number of times in the U direction based on the value provided.

Mirror: The texture is mirrored and then tiled.

Clamp: Restricts the texture look up to the range 0 to 1. The unmapped area is mapped with the last pixel value.

Border: Generates a border for the texturemap.

Mirroronce: Similar to Mirror, the texture is mirrored only once and not tiled.

Extended

Displays extended set of parameters of the respective material.

Default		
	Shine	0.500
	Edge	0.800
	Incident	0.900
	Fresnel	1.000
	Refractivity Index	1.030
	Shuffle Repeat	0
	Texture Cols	1
	Texture Rows	1
	Gamma	1.000
	Cubic Node	None
	Use Ambient Map	<input checked="" type="checkbox"/>
	Tag	

Default

- **Shine:** Defines the strength/intensity of the specular color on the surface of the object.
- **Edge:** Sets the intensity of refection from the front of the object,
- Incident: Sets the intensity of refection from the front of the object.
- Fresnel: Sets the blend between the reflection Min and Max.
- Refractivity Index: Specifies the refractivity Index of the texture applied in the refraction texture mode.
- Roughness: Specifies the roughness of the material when a roughness mode is selected for a texture in the texture layer.
- Shuffle Repeat: Specifies the phase value for he texture cell animation. If it is 0, It would not repeat, when 1 it starts to shuffle across the texture cells.
- Texture Cols: Specifies the vertical division in the texture of the material. This can be used to animate the material as a sprite.
- Texture Row: Specifies the horizontal division in the texture of the material. This can be used to animate the material as a sprite.
- **Gamma:** Sets Gamma properties to Increase or decrease the color intensity of the material.
- Cubic Node: Reference node for local cubic map.
- Use Ambient Map: Applies Ambient light according to the Environment map.

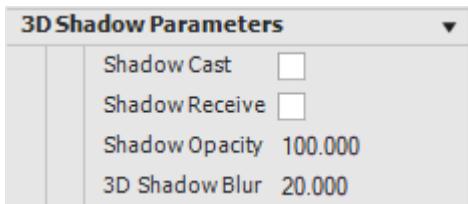
Faux Shadow Parameters

Faux Shadow Parameters	
	Faux Shadow <input type="checkbox"/>
	Object Visibility <input checked="" type="checkbox"/>
	Faux Shadow B... 0.000
	Blur Ratio 1.000
	Faux Shadow O... 0
	Faux Shadow A... 0
➤	Faux Shado... 0,0,0,200
	Shadow Depth 5.000

- **Faux Shadow:** Toggles visibility of shadow of the material
- Object Visibility: Toggles the visibility of the objects on which the selected material is applied.

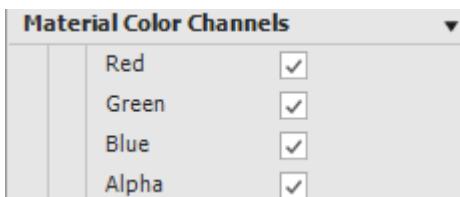
- **Faux Shadow Blur:** Sets the amount of blur of the faux shadow.
- **Blue Ratio:** Sets the scale of the blur of the faux shadow,
- **Faux Shadow Offset:** Sets the distance of shadow from the object shadow.
- **Faux Shadow Angel:** Sets the angle of refection in respect to the object.
- **Faux Shadow Color:** Sets the color and alpha of the shadow.
- **Shadow Depth:** Sets the shadow depth in respect to the object.

3D Shadow Parameters



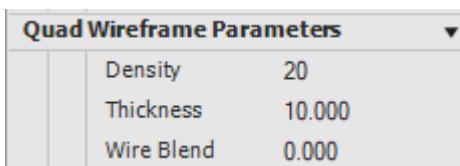
- **Shadow Cast:** Enables casting of shadow of the object to which the material is applied.
- **Shadow Receive:** Enables the object to receive 3D shadows on its surface to which the material applied.
- **Shadow Opacity:** Sets a normalized value for opacity of the casted shadow. Works on the receiver object.
- **3D Shadow Blue:** Sets a normalized value for the amount of blur of the casted shadow. Works on the receiver object.

Material Color Channels



- **Red:** When ON, renders the Red channel of the Material.
- **Green:** When ON, renders the Green channel of the Material.
- **Blue:** When ON, renders the Blue channel of the Material.
- **Alpha:** When ON, renders the Alpha channel of the Material.

Quad Wireframe Parameters



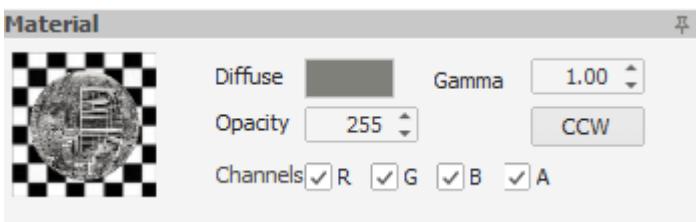
- **Density:** Sets the density of wire. Change fill mode to quad wire-frame to see the changes.
- **Thickness:** Sets the size of wire, Change fill mode to quad wire-frame to see the changes.
- **Wire Blend:** Sets the blend between fill and wire-frame of the object, change fill mode to quad wire-frame to see the changes,

Light Parameters

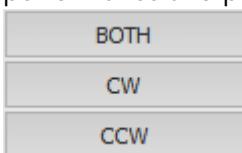


- Light Parameters: Specifies the light for the material. A separate light (other than default light) can be specified to affect the material.
- Sampler Type: Provides various texture blending modes, which can be used to blend between textures applied in different layers of the material.
- Render States: Provides options like fill mode, cull mode, point size and object blending parameters. Object blending works on the entire object to which the respective material is applied.

2. Material - Uniform

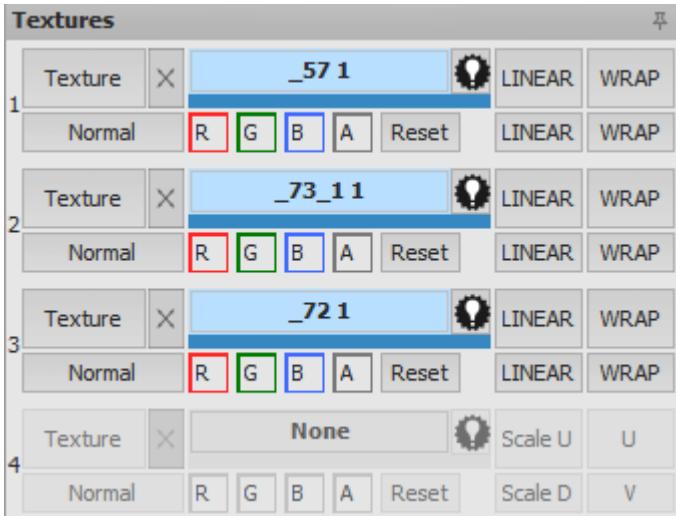


- Diffuse : The inherent surface color or texture of an object that evenly scatters light, determining its appearance under direct lighting.
- Gamma : A measure of brightness and contrast in digital imaging, affecting color accuracy and display.
- Opacity : The degree to which the color of an object or material is transparent or visible. Higher opacity means less transparency.
- Cullmode: Determines which side(s) of polygons are not rendered, optimizing rendering performance and preventing unnecessary calculations for unseen faces.



- BOTH: This term might refer to rendering or display modes where both front-facing and back-facing polygons or surfaces are visible and rendered. Typically, in many 3D engines or software, you can choose to render both sides of polygons (front and back) or only the front-facing side for optimization purposes.
- CCW: Abbreviation for "Counter ClockWise," often used in graphics to define the order of vertices that determine the orientation of a polygon face.
- CW: Stands for "Clockwise." In 3D graphics, it refers to the order in which vertices of a polygon are arranged to define the face of that polygon. When vertices are arranged in a clockwise order, it determines the front-facing side of the polygon.

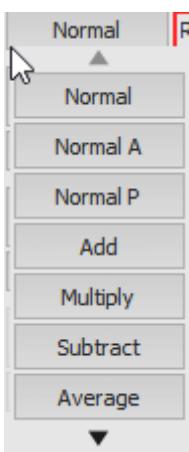
- Textures



Texture Type defines the behavior of texture when mixed with material.

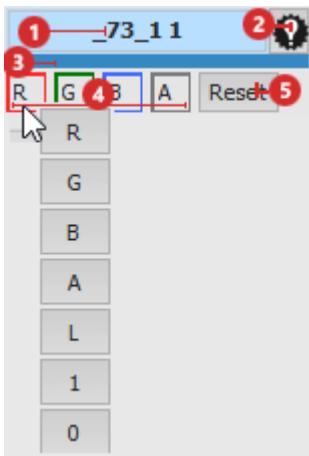


- Texture: The Texture option wraps a bitmap image onto an object's surface to create a visually realistic object. A bitmap consists of texture pixels or textures , where each texture provides the visual complexity which otherwise would have to be modeled. This advantage of lower polygon numbers helps reduce the rendering power while maintaining the frame-rate. X (X-Ray) option is available to enable X-Ray mode for texture when applied to a material of an object, the texture is scaled to the size of the viewport/render-area. The fill area of the object works as a window to reveal the texture.
- Opacity: Opacity is a kind of texture-map where the bitmap wrapped on the surface, is used to define the alpha (transparency) of the object surface. X option enables the selected texture to work in X-ray mode.
- Gradient: Gradient wipes the Luminance value of the selected channel.
- Mirror: The texture is mirrored and then tiled.
- Normal - Filters

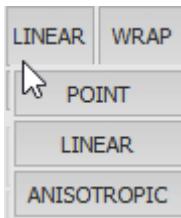


- Normal: The default mode, where no blending between texture maps occurs. Each texture pixel on the surface is painted as it appears in the original texture bitmap. Blending occurs between the base color and blend color based on the transparency of the texture pixels.
- Normal P: Works with Premultiplied Alpha, where the texture is blended with the layer behind it based on its alpha values. This mode assumes the colors in the texture have already been multiplied by the alpha, resulting in smoother blending, especially for semi-transparent textures.
- Normal A: In this mode, the alpha is multiplied separately, and colors are then blended based on their transparency. This method provides more control over how the colors interact with the layers behind, often leading to more precise edge blending.
- Add: This mode adds the base color to the blend color, resulting in a brighter color that moves towards white.
- Multiply: Multiplies the base color by the blend color, which results in a darker image. It is commonly used for shadow effects because multiplying any color by black results in black, while multiplying by white leaves the color unchanged.
- Subtract: Subtracts the blend color from the base color or vice versa, depending on which has a greater brightness value. This moves the result towards black.
- Average: Averages all the color components (RGBA) of the base and blend colors, resulting in a mid-tone between the two.
- Screen: Both the base and blend colors are inverted, multiplied by each other, and then the result is inverted again. Screening with black leaves the color unchanged, while screening with white results in white.
- Darken: Compares the base and blend colors and takes the darkest one as the output color.
- Lighten: Compares the base and blend colors and takes the lightest one as the output color.
- Exclusion: Produces a result where black and white return the same result, but colors in between become gray. This mode creates a "difference" effect but with lower contrast, and can be used to invert parts of an image.
- Overlay: A combination of multiply and screen modes, depending on the base color. If the base color is light, the blend color is screened. If the base color is dark, the blend color is multiplied.
- Hard Light: Uses the blend color as the base and either multiplies or screens the colors, depending on the blend color. If the blend color is lighter than 50% gray, the base is screened; if darker, it's multiplied.
- Soft Light: Similar to hard light but with a more subtle effect. It uses the blend color as the base and darkens or lightens the colors, adding a soft contrast to the image.
- Color Dodge: Brightens the base color to reflect the blend color by dividing the base by the inverse of the blend color. This creates a highlight effect.
- Color Burn: Darkens the base color to reflect the blend color by inverting, dividing, and inverting again. It creates a burn or shadow effect.
- Classic Color Dodge: Similar to Color Dodge but follows a more traditional color dodge algorithm, producing a smoother blend.
- Classic Color Burn: Similar to Color Burn but uses a traditional burn algorithm, giving a more controlled darkening effect.
- Linear Dodge: Similar to Color Dodge but uses linear interpolation, resulting in a smoother brightening effect.
- Echo: Adds shine or light to a texture. The output resembles soft light but with a more pronounced luminance effect.

- Glow: Similar to Echo but with the base and blend colors swapped. The result is closer to hard light, emphasizing highlights.
- Freeze: A variation of the Echo mode, where the base color is inverted, the blend color is inverted, and the result is inverted again, creating a cooling or frosted effect.
- Heat: Similar to Freeze but with the base and blend colors swapped, producing a warming effect rather than a cooling one.
- Interpolation: Similar to Average mode but with increased contrast between the base and blend colors.
- Stamp: The output is a blend of the base and blend colors, similar to Average, but works best with medium-toned colors, creating a stamped or embossed effect.
- Invert: Inverts the base color. Black becomes white, white becomes black, and other colors are converted to their complementary values.
-



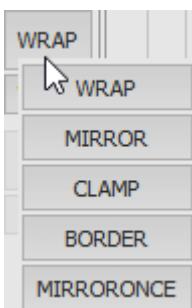
- Texture Name: Displaying the name of the texture.
- Inherit Scene Light: Implies objects in a 3D scene adopting the lighting characteristics of the overall scene rather than having individualized lighting settings.
- Opacity Slider: Manage the transparency levels of the texture.
- RGBA
 1. R: Represents the Red color channel in digital graphics, defining the intensity of red in a color.
 2. G: Represents the Green color channel, determining the intensity of green in a color.
 3. B: Represents the Blue color channel, indicating the intensity of blue in a color.
 4. A: Denotes the Alpha channel, controlling transparency or opacity in addition to color.
 5. **L:** Represents the Luminance channel, often used to describe the brightness or lightness of a color.
 6. **1:** Often signifies the maximum value or full intensity in certain color channels, such as in some color models where 1 represents the highest value.
 7. **0:** Often represents the minimum value or absence of intensity in certain color channels, such as in some color models where 0 signifies no intensity or absence of color.
- **Reset:** Resetting the RGBA color values back to their default settings.
- **Tag:** Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of he objects by specifying object. Tag as target.



Point: Color values of the closest adjacent pixel are taken to smoother the texture.

Linear: It is the default option & uses Bilinear Filtering in which a weighted average of four adjacent texture pixels is taken to create a single Texel.

Anisotropic: It takes into account the angle a primitive is rendered. Anisotropic compensates for distortion caused by the difference in angle between the texture polygon and the plane of the screen.



Wrap: Texture repeats n number of times in the U direction based on the value provided.

Mirror: The texture is mirrored and then tiled.

Clamp: Restricts the texture look up to the range 0 to 1. The unmapped area is mapped with the last pixel value.

Border: Generates a border for the texturemap.

Mirroronce: Similar to Mirror, the texture is mirrored only once and not tiled.

Extended

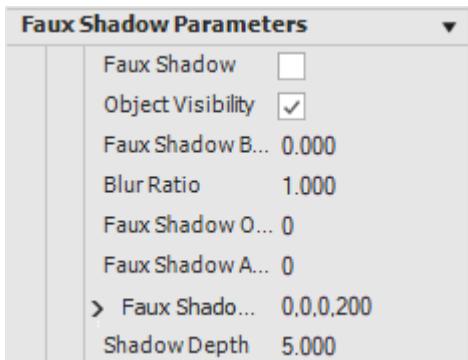
Displays extended set of parameters of the respective material.

Default		
	Shine	0.500
	Edge	0.800
	Incident	0.900
	Fresnel	1.000
	Refractivity Index	1.030
	Shuffle Repeat	0
	Texture Cols	1
	Texture Rows	1
	Gamma	1.000
	Cubic Node	None
	Use Ambient Map	<input checked="" type="checkbox"/>
	Tag	

Default

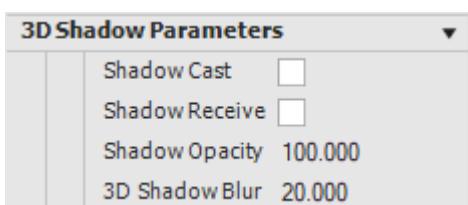
- **Shine:** Defines the strength/intensity of the specular color on the surface of the object.
- **Edge:** Sets the intensity of refection from the front of the object,
- Incident: Sets the intensity of refection from the front of the object.
- Fresnel: Sets the blend between the reflection Min and Max.
- Refractivity Index: Specifies the refractivity Index of the texture applied in the refraction texture mode.
- Roughness: Specifies the roughness of the material when a roughness mode is selected for a texture in the texture layer.
- Shuffle Repeat: Specifies the phase value for he texture cell animation. If it is 0, It would not repeat, when 1 it starts to shuffle across the texture cells.
- Texture Cols: Specifies the vertical division in the texture of the material. This can be used to animate the material as a sprite.
- Texture Row: Specifies the horizontal division in the texture of the material. This can be used to animate the material as a sprite.
- **Gamma:** Sets Gamma properties to Increase or decrease the color intensity of the material.
- Cubic Node: Reference node for local cubic map.
- Use Ambient Map: Applies Ambient light according to the Environment map.

Faux Shadow Parameters



- **Faux Shadow:** Toggles visibility of shadow of the material
- Object Visibility: Toggles the visibility of the objects on which the selected material is applied.
- Faux Shadow Blur: Sets the amount of blur of the faux shadow.
- **Blue Ratio:** Sets the scale of the blur of the faux shadow,
- **Faux Shadow Offset:** Sets the distance of shadow from the object shadow.
- **Faux Shadow Angel:** Sets the angle of refection in respect to the object.
- **Faux Shadow Color:** Sets the color and alpha of the shadow.
- **Shadow Depth:** Sets he Shadow depth in respect to the object.

3D Shadow Parameters



- **Shadow Cast:** Enables casting of shadow of the object to which the material is applied.
- **Shadow Receive:** Enables the object to receive 3D shadows on its surface to which the material applied.
- **Shadow Opacity:** Sets a normalized value for opacity of the casted shadow. Works on the receiver object.
- **3D Shadow Blue:** Sets a normalized value for the amount of blur of the casted shadow. Works on the receiver object.

Material Color Channels

Material Color Channels		
	Red	<input checked="" type="checkbox"/>
	Green	<input checked="" type="checkbox"/>
	Blue	<input checked="" type="checkbox"/>
	Alpha	<input checked="" type="checkbox"/>

- **Red:** When ON, renders the Red channel of the Material.
- **Green:** When ON, renders the Green channel of the Material.
- **Blue:** When ON, renders the Blue channel of the Material.
- **Alpha:** When ON, renders the Alpha channel of the Material.

Quad Wireframe Parameters

Quad Wireframe Parameters		
	Density	20
	Thickness	10.000
	Wire Blend	0.000

- **Density:** Sets the density of wire. Change fill mode to quad wire-frame to see the changes.
- **Thickness:** Sets the size of wire, Change fill mode to quad wire-frame to see the changes.
- **Wire Blend:** Sets the blend between fill and wire-frame of the object, change fill mode to quad wire-frame to see the changes,

Light Parameters

Light Parameters		
	Light	Directional Light 0

- **Light Parameters:** Specifies the light for the material. A separate light(other than default light) can be specified to affect the material.
- **Sampler Type:** Provides various texture blending modes, which can be used to blend between textures applied in different layers of the material.
- **Render States:** Provides options like fill mode, cull mode, point size and object blending parameters. Object blending works on the entire object to which the respective material is applied.

Image Texture



The Image Texture is used to apply static or animated image files to objects in the scene. It supports a range of common image formats, playback control for animated content, equirectangular projection, and sprite-based frame selection using Rows, Columns, and Depth.

Supported File Formats: **.bmp, .dds, .hdr, .jpg, .jpeg, .gif, .png, .tga, .webp, .jfif .gif**
(supports animated playback)

Parametric		
Default		
	File Name	D:\Desktop_12_08_2024\In...
	Width	1,401.000
	Height	1,080.000
	Play Control	0.0000,-1.0000,0.0000,6... <input type="button" value="..."/>
	Totalframes	1
	Equirectangular	<input type="checkbox"/>
	Tag	
	Rows	1
	Columns	1
	Depth	0
	Landscape	<input type="checkbox"/>
	Mip Map	<input type="checkbox"/>

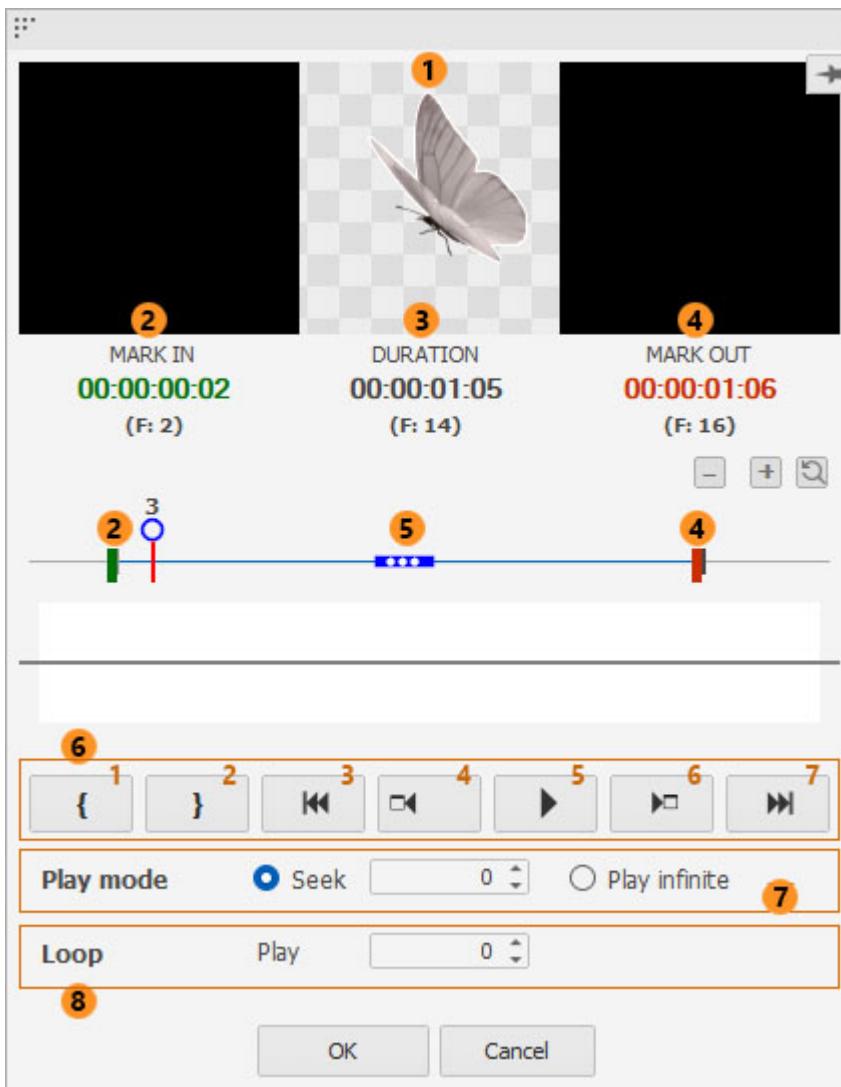
Parameters

File Name: Displays the path of the currently loaded image. Click the button to open the file dialog and change the image.

Width: Displays the width of the image in pixels. This is a read-only value.

Height: Displays the height of the image in pixels. This is a read-only value.

Play Control (Visible for animated formats like GIF): Used to control the playback range and frame navigation of animated images.



- 1. Image Frame Window:** Shows the video timeline and audio waveform.
- 2. Mark In:** Displays the frame number/time where playback would begin.
Displayed as: `00:00:00:00`
Frame: `F: 0` (default)
- 3. Duration:** Displays the total length of the animation.
Time: `00:00:SS:FF`
Frame: `F: [total frame count]`
- 4. Mark Out:** Displays the frame number/time where playback would stop.
Displayed as: `00:00:SS:FF`
Frame: `F: [last frame index]`
If Mark In and Mark Out are not set, the animation plays from the first to the last frame.
5. Clip Area Pointer (moves the duration segment between Mark In and Mark Out)
6. Playback Control Buttons
 1. Mark In: Sets the current frame as the Mark In point.
 2. Mark Out: Sets the current frame as the Mark Out point.
 3. Go to First Frame: Moves to the Mark In frame or the first frame.
 4. Previous Frame: Moves one frame backward.
 5. Play / Pause: Toggles playback.

6. Next Frame: Moves one frame forward.
7. Go to Last Frame: Moves to the Mark Out frame or the last frame
7. Playback Mode
 - Seek: Allows manual frame navigation.
 - Play Infinite: Plays the animation in a continuous loop.
8. **Loop Play** : Specifies how many times the animation should repeat. If the value is `0`, the user can choose between Seek and Play Infinite modes. If the value is greater than `0`, the animation will loop the specified number of times.

Total Frames: Displays the number of frames in the animated image.

Equirectangular : Toggles equirectangular projection mode. Used for rendering spherical or panoramic textures.

Tag: Stores a string value inside the texture. This can be used for reference using `sel f. tag` in action commands. It does not function as a dynamic variable.

Sprite Parameters

- **Rows:** Defines the number of horizontal divisions in the image. It is enabled only when the image is converted to a sprite. Default value: `1`.
- **Columns:** Defines the number of vertical divisions in the image. It is enabled only when the image is converted to a sprite. Default value: `1`.
- **Depth :** Specifies the index of the sprite cell to display. It is enabled only when the image is converted to a sprite. Valid range: `0` to `(Rows × Columns - 1)` Default value: `0`.

Landscape: Toggles the orientation of the texture between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mips to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Purpose of Mip Mapping: Mip maps are used to improve rendering quality and efficiency. They help reduce texture aliasing, shimmering, and more effects when textures appear small or are viewed at shallow angles. They also reduce GPU load by using optimized texture resolutions where possible.

Sprite Activation: The image is treated as a standard image by default.

To enable sprite-based indexing:

- Right-click the image and select Convert to Sprite
- Drag the image into the viewport, which automatically enables sprite controls.

Example: If `Rows = 2` and `Columns = 3`, total sprite cells = 6

- `Depth = 0` displays the first cell
- `Depth = 5` displays the last cell

DDS Texture



The DDS Texture is used to apply `.dds` format image files to objects in the scene. DDS (DirectDraw Surface) files may contain compressed, uncompressed, or mipmapped textures. These textures can represent standard 2D images or cubical images used for skyboxes and environmental mapping.

Supported File Format: `.dds`

Parametric		
Default		
	File Name	C:\Users\rabi.narayan\Down...
	Width	640.000
	Height	426.000
	Play Control	0.0000,-1.0000,0.0000,6630...
	Totalframes	1
	Equirectangular	<input type="checkbox"/>
	Tag	<input type="checkbox"/>
	Landscape	<input type="checkbox"/>
	Mip Map	<input type="checkbox"/>

Parameters

File Name : Displays the path of the currently loaded DDS image. Click the `...` button to open the file dialog and change the image.

Width: Displays the width of the texture in pixels. This is a read-only value taken from the file.

Height: Displays the height of the texture in pixels. This is a read-only value taken from the file.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Sprite Parameters (Visible only after conversion): DDS textures support optional sprite-based frame indexing, but **only for non-cubical DDS files**. If the DDS is cubical (e.g., used for cube maps or environment maps), sprite conversion does not apply. The following parameters are enabled only after the texture is converted to a sprite using the Convert to Sprite option:

Rows: Defines the number of horizontal divisions in the image. Default value: `1`.

Columns: Defines the number of vertical divisions in the image. Default value: `1`.

Depth: Specifies the index of the sprite cell to display. Valid range: `0` to `(Rows × Columns - 1)` Default value: `0`.

Sprite Activation : DDS textures are treated as standard 2D images by default. To enable sprite-based indexing (for non-cubical DDS files only):

- Right-click the DDS texture and select Convert to Sprite

Once converted, Rows, Columns, and Depth become available for sprite frame selection.

Note: `Sprite conversion is not applicable or effective for cubical DDS files.`

Example: If `Rows = 3` and `Columns = 2`, total sprite cells = 6

- `Depth = 0` shows the first cell
- `Depth = 5` shows the last cell

Video Texture



The Video Texture is used to assign video files as textures on objects in the scene. It supports playback control, audio waveform interaction, seeking, looping, and integration with client applications for remote playback triggers and behavior switching.

Parametric		
Default		
	File Name	C:\Users\rabi.narayan\Down...
	Width	640.000
	Height	360.000
	Total Frames	333
	Play Control	0.0000,-1.0000,0.0000,6630...
	Current Frame	0
	Decode Time	0.000
	Seek Buffers	2
	Audio Out	<input checked="" type="checkbox"/>
	Reload Video Stream	<input checked="" type="checkbox"/>
	Retry to Load(Times)	3
	Total Time	13.320
	Current Time	0.000
	Seekable	True
	Equirectangular	<input type="checkbox"/>
	Current Loop	0
	Tag	<input type="checkbox"/>
	Landscape	<input type="checkbox"/>
	Mip Map	<input type="checkbox"/>

Parameters (As Appearing in Interface Order)

File Name: Displays the path of the currently loaded video file. Click the  button to open the file dialog and select a new video.

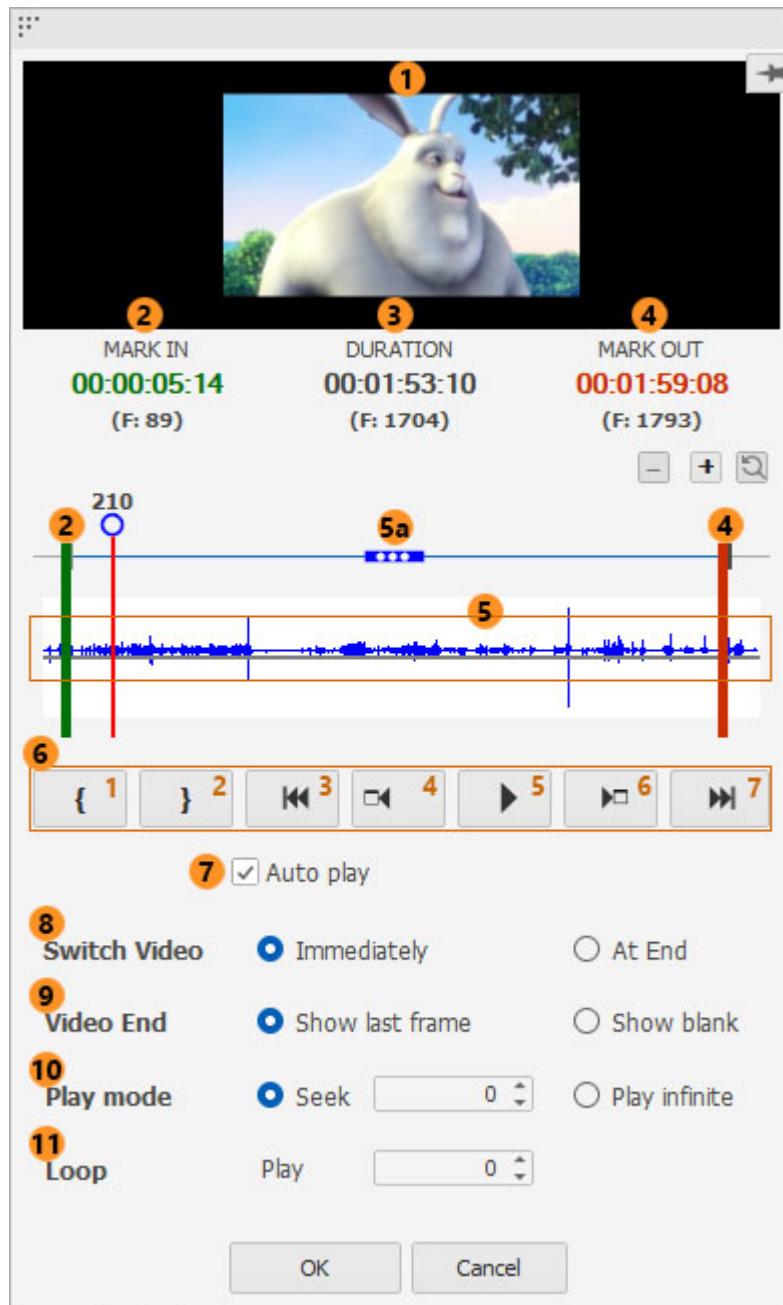
Width: Displays the width of the video in pixels. This is a read-only value based on the video file.

Height: Displays the height of the video in pixels. This is a read-only value based on the video file.

Total Frames: Displays the total number of frames in the video. This is a read-only value.

Play Control: The Play Control section enables timeline trimming, frame control, and playback

interaction. Video Frame Window - Shows the video timeline and audio waveform.



1. **Video Frame Window:** Shows the video timeline and audio waveform.
2. **Mark In:** Displays the frame and time where playback begins.
Format: `00:00:00:00`
Frame: `F: 0` (default)
3. **Duration:** Displays the total time and frame range between Mark In and Mark Out.
Format: `00:00:SS:FF`
Frame: `F: [frame count]`
4. **Mark Out:** Displays the frame and time where playback ends.
Format: `00:00:SS:FF`
Frame: `F: [last frame index]`
5. **Audio Waveform:** The waveform section allows interactive control over:
 - Mark In point

- Mark Out point
- Frame Header display
- (5a) Clip Area Pointer (moves the duration segment between Mark In and Mark Out)

6. Playback Control Buttons

1. Mark In: Sets current frame as Mark In
2. Mark Out: Sets current frame as Mark Out
3. Go to First Frame: Moves to Mark In or first frame
4. Previous Frame: Moves one frame backward
5. Play / Pause: Toggles playback
6. Next Frame: Moves one frame forward
7. Go to Last Frame: Moves to Mark Out or last frame

7. **Auto Play:** Toggle that determines whether the video should always start from the beginning when triggered On-Air. This is controllable only from the client application interface.

8. **Switch Video:** Radio buttons:

- Immediately
- At End

Specifies when a new video should start. Only controllable from the client application.

9. **Video End:** Radio buttons

- Show Last Frame
- Show Blank
- Determines what appears when the video ends.
- Only controllable from the client application.

10. Playback Mode

- Seek: Allows manual frame navigation
- Play Infinite: Loops the video indefinitely

11. **Loop Play:** Specifies how many times the video should repeat. If set to **0**, playback behavior depends on Playback Mode.

Current Frame: Displays the frame currently being played or viewed. This is a read-only value.

Decode Time: Displays the time taken to decode the current frame. This is a read-only value.

Seek Buffers: Specifies how many frames are buffered to improve seek performance. Higher values increase accuracy but may affect seeking speed.

Audio Out: Enables or disables audio output during playback. Reload Video Stream Manually reloads the video source. Used in cases where the stream has changed or failed to load.

Retry to Load (Times): Sets the number of retry attempts for loading the video stream.

Total Time: Displays the full length of the video in time format. This is a read-only value.

Current Time: Displays the current playback position in time format. This is a read-only value.

Seekable: Indicates whether the video is seekable (True or False). Only I-frame encoded videos are fully seekable. This is a read-only value.

Equirectangular: Toggles equirectangular projection for 360-degree videos.

Current Loop: Displays the number of times the video has looped during playback. This is a read-only value.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mirmapping:

- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

LIVE Texture



LIVE (Live Video): The Live Texture is used to apply live video input from supported capture devices to objects in the scene. It supports device-based selection, embedded audio toggling, and reading of ancillary data from the video signal.

Parametric	
Default	
Input Selection	Live 2
Device	DeckLink
Embedded Audio	<input checked="" type="checkbox"/>
Anc Data	
Tag	
Landscape	<input type="checkbox"/>
Mip Map	<input type="checkbox"/>

Parameters

Input Selection : Displays input options ranging from Live 1 to Live 8. The available options depend on the capture hardware installed on the machine. Each live input corresponds to a physical port on the device.

The maximum supported input count per device is 8.

Device: Displays the name of the video capture device associated with the selected input. Device preferences and port configurations can be managed through the ServerDef utility.

Embedded Audio: Checkbox to enable or disable audio output if the input signal includes embedded audio. When checked, audio is extracted from the live signal. When unchecked, no audio is played.

Anc Data: Displays any available Ancillary Data present in the live signal (such as captions, timecode, etc.). This is dependent on signal compatibility and device capability.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mips to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Snapshot Texture



SNP (Snapshot): The Snapshot Texture is used to display a real-time rendered frame from the scene or from a selected canvas. This allows capturing and reusing outputs within the same scene or across different layers of the production setup.

Parametric		
Default		
	Canvas	Default
	Tag	<input type="checkbox"/>
	Landscape	<input type="checkbox"/>
	Mip Map	<input type="checkbox"/>

Parameters

Canvas: Allows selection of a canvas from Canvas 1 to Canvas 8. Each canvas corresponds to an output render surface of the engine or a defined rendering view in the scene. When a canvas is selected, the Snapshot Texture displays its live frame content.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Render Texture



RT (Render Texture): The Render Texture allows rendering specific objects or object groups into a texture, functioning as a dedicated render buffer. It can be used to simulate reflections, apply secondary camera views, or composite multiple layers of a scene. Render Texture works with selected objects or groups. Only the objects assigned to the render texture are included in its output.

Parametric		
Default		
	RT Width	1,920.000
	RT Height	1,080.000
	Reflection	<input type="checkbox"/>
	Camera	Active
>	Clear Color	0,0,0,0
	Mirror Plane	None
	Cubic	<input type="checkbox"/>
	Use Cache	<input type="checkbox"/>
	Eye Point	None
	Inherit Backbuffer Size	<input type="checkbox"/>
	Render Target	Default
	Accumulate	0.000
	Matte Node	None
	Override Alpha	<input checked="" type="checkbox"/>
	Tag	<input type="checkbox"/>
	Landscape	<input type="checkbox"/>
	Mip Map	<input type="checkbox"/>

Parameters

RT Width: Sets the horizontal render resolution in pixels.

RT Height: Sets the vertical render resolution in pixels.

Reflection: When enabled, mirrors the rendered output. Commonly used for floor reflections.

Camera: Defines which camera should be used to render the scene into the texture. If not specified, the active viewport camera is used.

Clear Color: Sets the background color (excluding alpha) for the render output.

Mirror Plane: Specifies the deflector plane used to calculate planar reflections.

Cubic: When enabled, renders the output as a cubic map, allowing it to be used in environmental mapping scenarios.

Use Cache: Specifies where the render output is cached. Options:

- Default: Cache is local to the current scene.
- Cache 1 / Cache 2 / Cache 3: These buffers are shared globally across scenes in the Sting Server. A render done in one scene using Cache 1 can be reused in another scene that references the same cache slot.

Eye Point: Specifies an object to define the eye-level for perspective rendering.

Inherit Backbuffer Size: When enabled, the render texture will inherit the resolution of the active backbuffer (viewport size) when the scene is loaded in various resolutions.

Render Target: Selects the render target surface for output.

Accumulate: Determines how many previous frames are retained in the output.

- **0** : No frame accumulation
- **100** : All frames persist
- Any intermediate value: Frames stay temporarily based on the set value and fade out gradually. Useful for motion trails or temporal blending.

Matte Node: Specifies a node to be used as a matte layer. Typically, this is a deflector plane used to mask or clip portions of the reflection or render output.

Override Alpha: When enabled, the alpha of the render output is used in the final texture.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through **self.tag**.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Brightness and Contrast Texture



BC (Brightness/Contrast): It provides basic color correction controls.

Parametric		
Default		
	Hue RotationAngle	1.000
	Saturation	1.000
	Brightness	1.000
	Contrast	1.000
>	Threshold	1.000,50.000,100.000,0.000
>	Output Levels	0.000,100.000
	Tag	
	Landscape	<input type="checkbox"/>
	Mip Map	<input type="checkbox"/>

Parameters

Hue RotationAngle: Shifts the relative Hue color range of the texture,

Saturation: Sets the saturation of the material and texture layers above the Brightness Contrast Texture.

Brightness: Sets the brightness of the material and texture layers above the Brightness Contrast Texture.

Contrast: Sets the contrast of the material and texture layers above the Brightness Contrast Texture.

Threshold: Modifies the Shadow, Midtone and Highlights of the Material

Output Levels: Modifies the Black and White level of the material.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mips to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Browser Texture



Browser: Allows inserting web link.

Parametric		
Default		
	URL	wasp3d.com
	Width	1,920.000
	Height	1,080.000
	Tag	
	Landscape	<input type="checkbox"/>
	Mip Map	<input type="checkbox"/>

Parameters

URL: Specifies the URL to display in the Browser texture

Width: Displays the Horizontal size of the texture in Pixels.

Height: Displays the Vertical size of the texture in Pixels.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

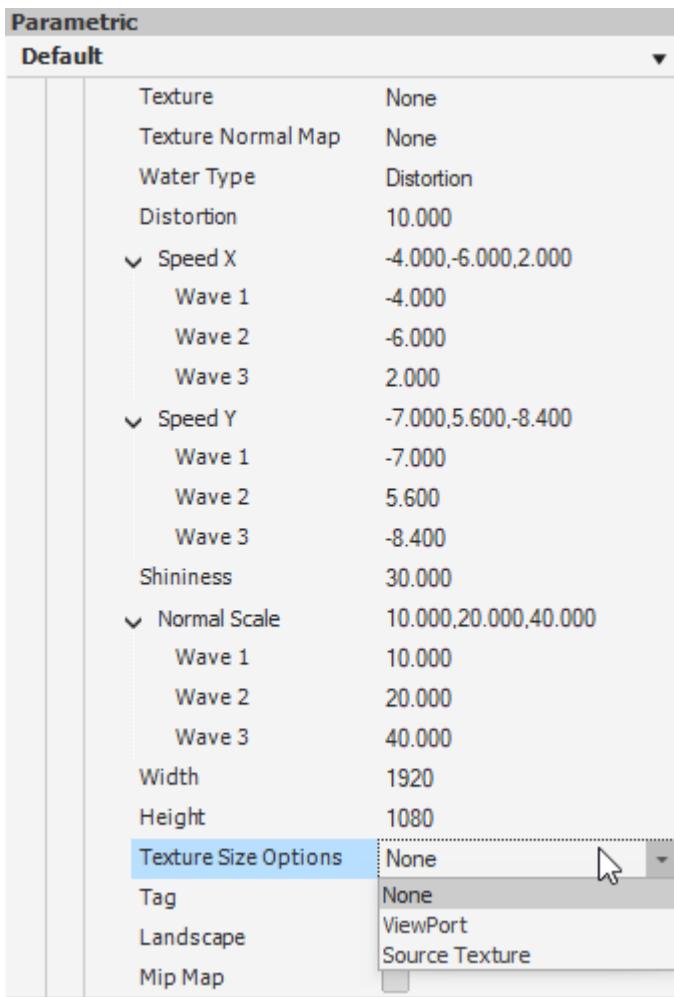
Mip Map: Enables or disables mipmapping:

- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Water Texture



The Water Texture simulates animated wave surfaces using a normal map-based distortion technique. It supports configuration of wave speed, direction, distortion intensity, and visual surface properties across three independent wave layers. A default normal map is applied automatically, but users can assign a custom one through the Texture Normal Map parameter.



Parameters

Texture: Specifies the base texture to be applied to the water surface

Texture Normal Map: Overrides the default normal map with a custom one for defining wave direction and distortion detail.

Water Type: Defines the water behavior mode, depending on the internal shader logic.

Distortion: Controls how strongly the normal map distorts the base texture.

Wave Speeds:

Speed X

- Wave 1: Horizontal speed of the first wave layer.
- Wave 2: Horizontal speed of the second wave layer.
- Wave 3: Horizontal speed of the third wave layer.

Speed Y

- Wave 1: Vertical speed of the first wave layer.
- Wave 2: Vertical speed of the second wave layer.

- Wave 3: Vertical speed of the third wave layer.

Shininess: Controls the intensity of specular highlights on the water surface.

Normal Scale

- Wave 1: Controls the distortion scale of the first wave layer.
- Wave 2: Controls the distortion scale of the second wave layer.
- Wave 3: Controls the distortion scale of the third wave layer.

Width: Displays the horizontal size of the texture in pixels. (Read-only)

Height: Displays the vertical size of the texture in pixels. (Read-only)

Texture Size Options: Defines how the size of the Water Texture is determined:

- None: Uses the render area size at the time the Water Texture is created.
- Viewport: Matches the size of the current viewport or default render area.
- Source Texture: Matches the size of the texture specified in the Texture parameter.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Blur Texture

B The Blur Texture enables various screen-space post-processing effects through a selectable Shader Type. Each shader type uses a set of parameters that define how the source texture is processed to achieve the desired visual effect.

Parametric		
Default		
	Shader Type	Gaussian Blur
	Source Texture	
	Blur Mode	Blur 2
	Contrast	1.000
›	Blur Params	0.000,0.000,50.000,50.000
	Width	1920
	Height	1080
	Texture Size Options	None
	Tag	
	Landscape	<input type="checkbox"/>
	Mip Map	<input type="checkbox"/>

Parameters (Common to All Shader Types)

- **Shader Type:** Selects the blur or visual effect shader.
- **Source Texture:** Defines the input texture to be processed.
- **Width:** Displays or sets the width of the output texture.
- **Height:** Displays or sets the height of the output texture.
- **Tag:** Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.
- **Landscape:** Toggles the orientation of the texture display between portrait and landscape.
- **Mip Map:** Enables or disables mipmapping:
 - Enabled: Uses mips to optimize texture scaling and performance at various distances or resolutions.
 - Disabled: Displays the original resolution of the texture at all times.

Other parameters vary depending on the selected Shader Type, as detailed below:

Shader Type: Gaussian Blur

- **Blur Mode:** Specifies blur kernel size – options include Blur 1, 2, 4, 8, 16, 32.
- **Contrast:** Adjusts contrast of the result.
- **Blur Params:** Defines custom parameters for the blur (shader-specific use).
- **Blur Size:** Controls the intensity or spread of the blur.
- **Angle:** Specifies blur angle in degrees (used in directional contexts).
- **Position X (%):** X-position in percent for applying the blur.
- **Position Y (%):** Y-position in percent for applying the blur.
- **Texture Size Options:** Determines how the output size is derived.

Shader Type: Depth of Field (DOF)

- **DOF Params:** Internal shader parameters for DOF behavior.
- **Blur Size:** Defines the blur amount in out-of-focus regions.
- **Focus Distance:** Distance from the camera where the image is sharpest.

- F-Stop: Aperture value controlling the depth of field.
- Focal Point: Target point in the scene for focus.

Shader Type: Directional Blur / Spin Blur / Zoom Blur

- Blur Mode: Kernel size selection (as in Gaussian Blur).
- Contrast: Controls contrast of the result.
- Blur Params: Shader-specific blur parameters.
- Blur Size: Strength of blur effect.
- Angle: Direction of motion or spin blur.
- Position X (%): X-origin for the effect.
- Position Y (%): Y-origin for the effect.
- Texture Size Options: Output sizing logic

Shader Type: God Rays

- Blur Params: Internal shader controls.
- Blur Size: Defines ray spread.
- Length: Controls the stretch or depth of rays.
- Position X (%): Horizontal origin of rays.
- Position Y (%): Vertical origin of rays.

Shader Type: Glow Blur

- Blur Mode: Kernel size control.
- Contrast: Contrast control.
- Blur Params: Additional blur tuning values.
- Blur Size: Size or softness of the glow.
- Angle: Glow spread direction.
- Position X (%): Glow X-origin.
- Position Y (%): Glow Y-origin.
- Glow Threshold: Brightness threshold for glow to activate.
- Texture Size Options: Texture sizing logic.

Shader Type: SSAO (Screen Space Ambient Occlusion)

- SSAO Params: Internal AO parameter group.
- Intensity: Strength of occlusion.
- Bias: Self-occlusion reduction threshold.
- Sample Radius: Radius used to compute surrounding samples.
- Scale: Overall scale of the SSAO effect.

Shader Type: Lens Distort

- Distort Params: Internal lens distortion control.
- Center Pos X: X-origin of distortion.

- Center Pos Y: Y-origin of distortion.
- K1: Radial distortion coefficient.
- K2: Higher-order radial distortion coefficient.

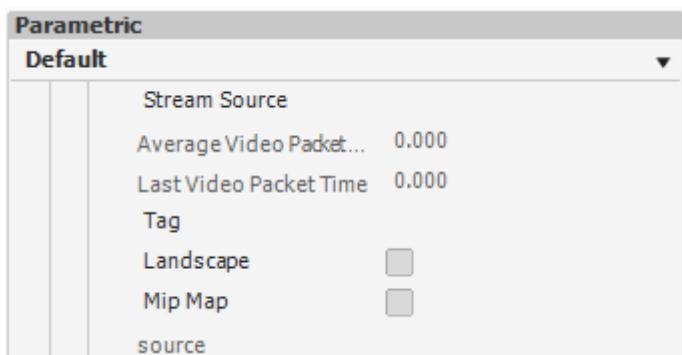
Shader Type: Lens Undistort

- Undistort Params: Internal undistortion parameters.
- Center Pos X: Center of undistortion (X-axis).
- Center Pos Y: Center of undistortion (Y-axis).
- K1: First distortion correction coefficient.
- K2: Second distortion correction coefficient.
- Sampler X: X-axis sampling scale.
- Sampler Y: Y-axis sampling scale.
- Border Color: Color used to fill empty areas post-undistortion.
- Source Texture Matrix: Matrix defining how to transform the source texture.

Netwrok Stream Texture



Netwrok Stream Texture: The NDI Texture captures a live video feed from a selected NDI® source and applies it as a texture within the scene. Once captured, the feed can be mapped to objects, used within material layers, or integrated into compositions, enabling realtime input from an NDI input.



Parameters

Stream Source: Selects the NDI input port from NDI 1 to NDI 8, based on the source assignments in the system configuration.

Average Video Packet Time: Displays the average time taken to receive video packets from the selected NDI source.

Last Video Packet Time: Displays the time taken to receive the video packet corresponding to the most recently rendered frame.

Source: Read-only field that shows the name of the NDI stream currently assigned to the selected input port.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Remap Texture



Remap Texture: The Remap Texture enables recoloring of an image by remapping tonal values through a multi-color gradient. It is especially useful for creating stylized visual effects by applying color tones to specific luminance or channel-based ranges of an image.

Parametric		
Default		
	Channel Type	Luma
	Color1	 255,0,0,255
	Red	255
	Green	0
	Blue	0
	Alpha	255
	Color2	 0,255,0,255
	Color3	 0,0,255,255
	Color4	 255,255,0,255
	Colors Steps	0.000,30.000,60.000,100.000
	X	0.000
	Y	30.000
	Z	60.000
	W	100.000
	Center Position	0.000,0.000,0.000,1.000
	Rotation Angle	0.000
	Tag	
	Landscape	
	Mip Map	

Parameters

Channel Type: Defines the source data on which the color remap is applied. This channel is used to evaluate the intensity across the texture for gradient-based mapping.

Options include:

- Red: Uses the red channel intensity of each pixel as the remapping source.
- Green: Uses the green channel intensity of each pixel as the remapping source.
- Blue: Uses the blue channel intensity of each pixel as the remapping source.
- Luma: Calculates the luminance of the pixel using a weighted average (commonly $Y = 0.299R + 0.587G + 0.114B$) and uses that value for remapping.

The selected channel determines how the gradient is applied across the image based on pixel intensity.

Color Mapping (Color1 – Color4) : You can define up to four color stops in the gradient. These colors are used to remap the selected input channel's range.

Each color includes the following components:

Color1

- Red: Red intensity (0–255)
- Green: Green intensity (0–255)
- Blue: Blue intensity (0–255)
- Alpha: Opacity (0–255)

Color2

- Same as Color1

Color3

- Same as Color1

Color4

- Same as Color1

These four colors form the gradient. When the remap is applied, pixels with lower intensity values will start from Color1, midrange values transition through Color2 and Color3, and high intensity values map to Color4.

Color Steps (X, Y, Z, W)

These define the position of each color in the gradient on a percentage scale. The values range from 0 to 100:

- X: Position of Color1 in the gradient.
- Y: Position of Color2
- Z: Position of Color3
- W: Position of Color4

The positions determine how quickly or slowly the gradient transitions between colors based on the selected channel's values.

Center Position (X, Y, Z, W)

Defines the center blending position between each pair of adjacent gradient colors. This fine-tunes the curve of transition between each color.

- X: Center between Color1 and Color2
- Y: Center between Color2 and Color3
- Z: Center between Color3 and Color4
- W: Reserved (may serve future or advanced gradient blending control)

Rotation Angle: Specifies the rotation of the gradient color chart in terms of hue orientation. When applied, this modifies the direction in which the gradient colors are evaluated across the image—useful for radial or angular tone-based remaps.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mips to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

LensFlare Texture



The LensFlare Texture simulates a multi-ring optical lens flare commonly seen in cameras and cinematic rendering. It consists of a central source and up to four colored rings that mimic realistic lens light scattering.

Global Parameters

Multiplier: Multiplies the overall color intensity of the entire lens flare setup. Useful to globally brighten or dim all ring elements at once.

Brightness: Adjusts the base brightness of the central flare without affecting ring intensities.

Source Texture: Specifies the bitmap/texture used to draw the lens flare. Chosen from the Texture Pool.

Reference Node: Assigns an object whose position defines the base point of the flare in 3D space.

- Animating this node moves the flare origin accordingly.
- The node can be hidden after use if not part of the final design.
- Think of it as the virtual light source that determines lens flare direction and anchor.

Ring Controls

Each ring has its own control set. The flare is split into four independent rings - Ring 1, Ring 2, Ring 3, and Ring 4. Each ring includes the following:

Intensity (R, G, B): Defines the brightness of the individual red, green, and blue color channels for the ring.

- R (Red Intensity): Sets the intensity of the red color in the ring.
- G (Green Intensity): Sets the intensity of the green color in the ring.
- B (Blue Intensity): Sets the intensity of the blue color in the ring.

A higher value means the color will appear more prominently in the ring.

Radius (R, G, B): Specifies how far each color channel sits from the center of the lens flare.

- R (Red Radius): Distance of the red channel from the flare's center.
- G (Green Radius): Distance of the green channel from the flare's center.
- B (Blue Radius): Distance of the blue channel from the flare's center.

Use slightly different values for R, G, and B to simulate chromatic dispersion.

Distance: Defines how deep into the "lens" the ring appears to be rendered. It controls parallax and layering for each ring.

- Distance: Determines the visual layering depth of the ring. Higher values push the ring farther away from the source node in visual space.

Per Ring Layout

Ring 1

- Intensity 0 – R, G, B
- Radius 0 – R, G, B
- Distance 0

Ring 2

- Intensity 1 – R, G, B
- Radius 1 – R, G, B
- Distance 1

Ring 3

- Intensity 2 – R, G, B
- Radius 2 – R, G, B
- Distance 2

Ring 4

- Intensity 3 – R, G, B
- Radius 3 – R, G, B
- Distance 3

Additional Parameters

- **Tag:** User-defined tag for organizational or scripting purposes.
- **Landscape:** A toggle used when rendering scenes in landscape orientation.
- **Mip Map:** Controls texture mipmapping for the lens flare texture, improving texture quality at varying distances or sizes.

✓ Usage Tips

- Animate the Reference Node to simulate moving headlights or sun glints.
- Use stronger R or B values in outer rings to create vibrant outer fringes.
- Vary Distance per ring to add realism through light scattering depth.

Let me know if you want this version formatted for GitHub markdown, internal help doc format, or part of your ongoing user manual.

Gradient Texture



Gradient: A custom texture to create Gradient effect using four color gradients in four gradient types (Rectangle, Radial, Conical, linear).

Parametric		
Default		
	Gradient Type	Rectangle
	Width	1920
	Height	1080
	> Color1	 255,0,0,255
	> Color2	 0,255,0,255
	> Color3	 0,0,255,255
	> Color4	 255,255,0,255
	> Colors Steps	0.000,30.000,60.000,100.000
	> Center Position	0.000,0.000,0.000,1.000
	Rotation Angle	0.000
	Heatmap Rows	2
	Heatmap Columns	2
	Heatmap Data	<table><data datasource="">
	Heatmap Colors	<table><data datasource="">
	Pixel Radius	6.000
	Blur	0.500
	Radial Segments	3
	Centered	<input type="checkbox"/>
	Tag	<input type="checkbox"/>
	Landscape	<input type="checkbox"/>
	Mip Map	<input type="checkbox"/>

Parameters

Gradient Type : Specifies the pattern of the gradient distribution across the texture.

- Rectangle: Applies each color (Color1 to Color4) to one of the four corners of the texture. The gradient is formed by blending between these corner color points.
- Radial: Creates a circular gradient from the center outward, based on the defined colors and their step positions.
- Conical: Forms an angular sweep of color transitions centered around the middle of the texture.
- Linear: Applies a directional gradient in a straight line from one edge to another. The direction is controlled using the Rotation Angle.
- Heatmap: Generates a grid-based intensity map using external data values and color interpolation.
- Vertex: Applies colors based on the geometry's vertex attributes. The colors blend depending on how the geometry is shaped and subdivided.

Width: Specifies the horizontal resolution of the texture in pixels (X-axis). Example: `width = 512`

Height: Specifies the vertical resolution of the texture in pixels (Y-axis). Example: `height = 512`

Color1: Represents the first color stop in the gradient.

- Color1.Red: Red component (0–255)

- Color1.Green: Green component (0–255)
- Color1.Blue: Blue component (0–255)
- Color1.Alpha: Opacity value (0–255)

Color2: Represents the second color stop in the gradient.

- Color2.Red: Red component (0–255)
- Color2.Green: Green component (0–255)
- Color2.Blue: Blue component (0–255)
- Color2.Alpha: Opacity value (0–255)

Color3: Represents the third color stop in the gradient.

- Color3.Red: Red component (0–255)
- Color3.Green: Green component (0–255)
- Color3.Blue: Blue component (0–255)
- Color3.Alpha: Opacity value (0–255)

Color4: Represents the fourth color stop in the gradient.

- Color4.Red: Red component (0–255)
- Color4.Green: Green component (0–255)
- Color4.Blue: Blue component (0–255)
- Color4.Alpha: Opacity value (0–255)

Color Steps: Defines the position of each color stop within the gradient on a scale from 0 to 100.

- X: Position of Color1 in the gradient.
- Y: Position of Color2 in the gradient.
- Z: Position of Color3 in the gradient.
- W: Position of Color4 in the gradient.

Center Position: Controls the midpoint blending transition between each pair of adjacent colors.

- X: Midpoint between Color1 and Color2.
- Y: Midpoint between Color2 and Color3.
- Z: Midpoint between Color3 and Color4.
- W: Reserved for extended midpoint control.

Rotation Angle: Specifies the rotation of the gradient in degrees. This affects the direction in Linear, Radial, and Conical gradients. Example: `Rotation Angle = 90 rotates` the gradient 90 degrees clockwise.

Heatmap Parameters: (Visible only when Gradient Type is set to Heatmap)

- Heatmap Rows: Number of vertical divisions (grid rows) in the heatmap.
- Heatmap Columns: Number of horizontal divisions (grid columns) in the heatmap.

- Heatmap Data: Input for external data values used to populate the heatmap.
- Heatmap Colors: The color set or scheme used for data value mapping.
- Pixel Radius: Radius of influence per data point, in pixels.
- Blur: Smoothing factor for blending data point colors.

Radial Segments: (Used with Radial and Conical types) - Specifies how many radial segments the gradient should have. A higher number results in a smoother circular gradient.

Centered: A toggle to align the gradient pattern to the center of the texture.

- Enabled: Gradient origin aligns to the center.
- Disabled: Gradient origin aligns with the default or top-left.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

UBuffer Texture



The UBuffer Texture allows integration of rendered output and data buffers from Unreal Engine. It supports importing both color and depth buffers from Unreal's rendering pipeline. The texture can be used for visual rendering, compositing, or real-time interaction based on Unreal content. Camera data from the VSet Add-in is used to control the Unreal Engine camera dynamically, ensuring frame-accurate synchronization.



Parameters

Unreal Alpha: A toggle to enable or disable the alpha channel present in the Unreal color texture.

- Enabled: The alpha component is respected and imported along with the RGB data.
- Disabled: The alpha channel is ignored; only RGB content is considered.

Tag: Serves as a variable place holder to hold values for the object. user can specify object names in Tag and wire the equivalent properties of the objects by specifying Object. Tag as Target.

Landscape: Changes orientation of the texture

Buffer Type: Specifies the type of Unreal Engine rendering buffer to be used in the texture.

- Color Buffer: Captures the final rendered RGB output from Unreal Engine. This includes all lighting, material, and post-processing effects as visible in the main render.
- Depth Buffer: Captures the per-pixel depth information from the Unreal Engine's rendering stream. This buffer allows depth-based interaction between objects rendered in Unreal and elements present in the Wasp3D scene. It is primarily used to determine spatial relationships, such as which object appears in front or behind, enabling seamless depth compositing between the Unreal content and Wasp3D scene elements.

Mip Map: Enables or disables mirmapping:

- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Placeholder Texture

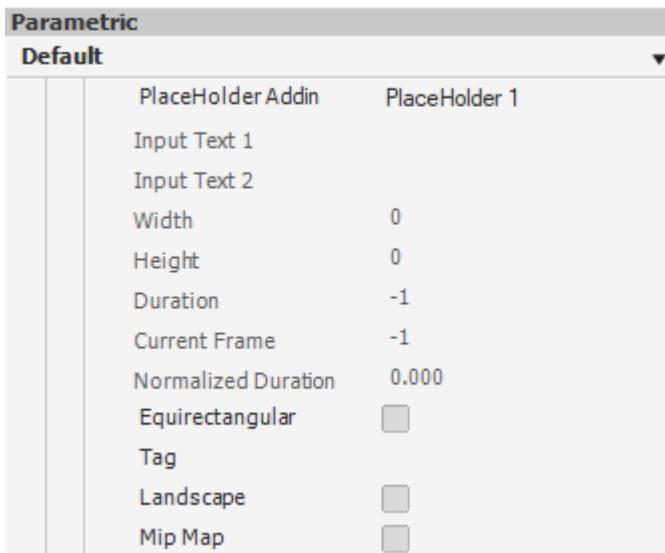


The Placeholder Texture allows dynamic assignment of various input types—such as live video, NDI feeds, still images, videos, or input from supported applications like Zoom—to an object in the scene. This texture type provides a flexible mechanism to change visual input without modifying scene wiring or logic.

Users can control which content is shown using the Placeholder control interface available in the Playout Client, Fluid App, or VSet App.

- Fluid App and VSet App provide access to all 8 placeholder channels.
- Playout Clients support Placeholder 1 only.

At any given time, up to 8 placeholders can be active in a scene. Placeholders are engine-level constructs, so changing one placeholder affects all objects using that specific channel.



Parameters

Place Holder Addin: This section allows users to define the placeholder channel and configure optional input metadata.

- Placeholder Channel: Select a placeholder from 1 to 8.
- Input Text 1: User-defined text for use in dynamic content logic (e.g., caption, label).
- Input Text 2: Additional user-defined text (optional).

Width: Displays the horizontal resolution (in pixels) of the content assigned to the placeholder.

Height: Displays the vertical resolution (in pixels) of the assigned input.

Duration: Total duration of the content, applicable if the input is a video or animated sequence.

Current Frame: Indicates the current frame being displayed from the selected input (if video or image sequence).

Normalized Duration: Represents the playback progress as a floating-point value between 0 and 1. This allows normalized animation or time-based logic.

Equirectangular: Toggle to enable support for 360° panoramic mapping. When enabled, the placeholder texture is treated as an equirectangular projection.

Tag: Stores a static string value that can be associated with the texture. This tag can be referenced in logic or scripting through `self.tag`.

Landscape: Toggles the orientation of the texture display between portrait and landscape.

Mip Map: Enables or disables mipmapping:

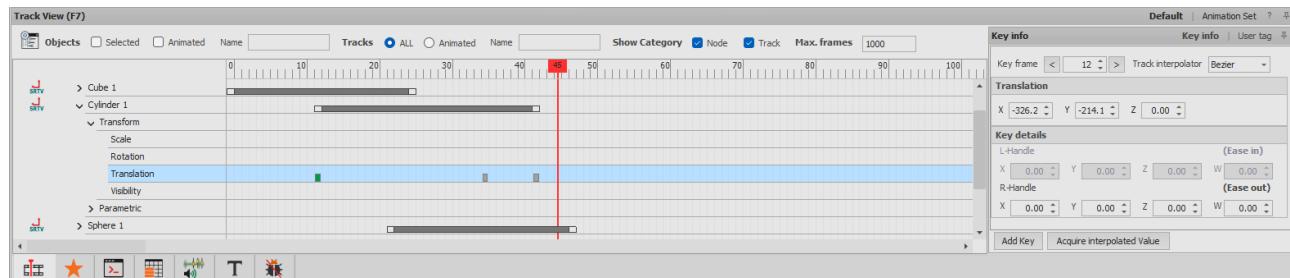
- Enabled: Uses mipmaps to optimize texture scaling and performance at various distances or resolutions.
- Disabled: Displays the original resolution of the texture at all times.

Track View (F7)

The Track View (F7) window displays timeline/track controls, key frames etc. for each scene object (node) and all its available transformations and parametric properties. It also provides controls to specify interpolation method for a particular track like Linear, Bezier, etc.

Do any of the following to open this window:

- On the Window menu, click Track View. Or
- Press (F7) from the keyboard.



Following are the detail in reference to the illustration of Parameters window:

1. Default: It is the scene's timeline. It contains all the objects of the scene with their interpolatable tracks. User can optionally filter the view by using Filter Mode.
2. Animation Set: It provides access to all animation sets created in the scene. Animation set is a pre-made animation of an interpolatable track. This track can be applied to an object on runtime and even multiple times based on certain events or triggers. These tracks are independent and apply over and above the respective object's native animation.

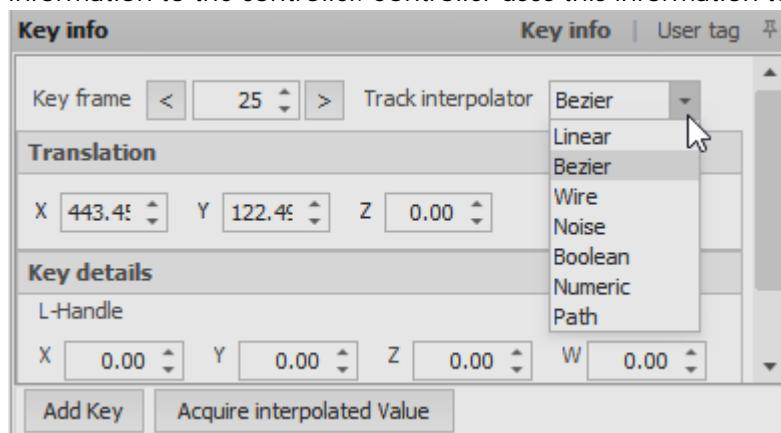
Key Frame values can be edited from within the Track View. Depending on the interpolation method selected, any additional properties like Bezier curve handles can also be edited within the same interface.

1.
 - **Objects (Materials Filter)** : The filters provided in this area apply to objects and Materials by their name.
 - Selected: When checked, only shows the tracks of object which is/are selected.
 - Animated: When checked, only shows the tracks of objects which is/are animated.
 - Name: User can filter the objects by their name. For e.g. "N*" will filter all the objects starting with "N". User can also write the complete name of the object to filter.
 - **Track (Filter)**: The filters provided in this area apply to objects and Materials by their name.

- All: Shows all the tracks (Transform & Parametric) in the Track view.
- Animated: When Checked, only shows the tracks (Transform & Parametric) of objects which is/are animated
- Name: User can filter the objects by their name. For e.g. "N*" will filter all the objects starting with "N". User can also write the complete name of the object to filter.
- Show Category: Tracks are grouped in the Trackview with under Node and Object based category. Textures, Materials, Objects, Camera all are called Nodes. Objects are further classified under Parametric and Transform tracks.
 - Node: Toggles track grouping as per Node.
 - Track: Toggles track grouping as per Object's Transform and Parametric track categories.
- Max Frames: Allows setting the number of maximum frames in the Scene Graph. By default, it is set to '1000' frames

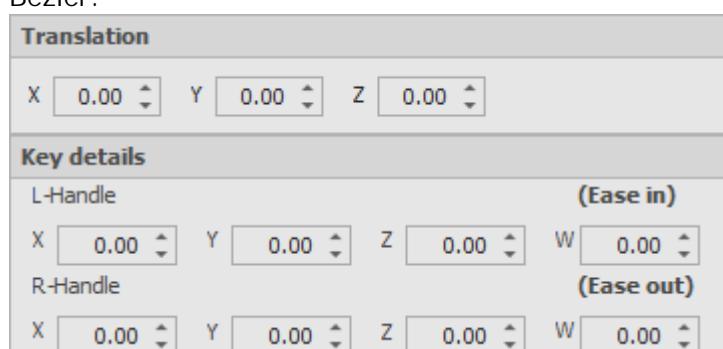
2. **Key Info** - Key frame values can be edited from within the TRACK VIEW and depending on the interpolation method selected, any additional properties like Bezier curve handles can also be edited within the same interface.

- Key Frame: It shows the editable frame number of the selected key frame. Modifying this changes, the frame value of the key frame.
- Track Interpolator: It stores and controls information of the keys and supplies this information to the controller. Controller uses this information to control the objects.



There are following types of Track Interpolator:

- Linear: Linear interpolation is a commonly used method for achieving smooth and consistent transitions between key frames in animations. With linear interpolation, the object's animation progresses evenly and consistently from one key frame to the next, maintaining a constant rate of change.
- Bezier:

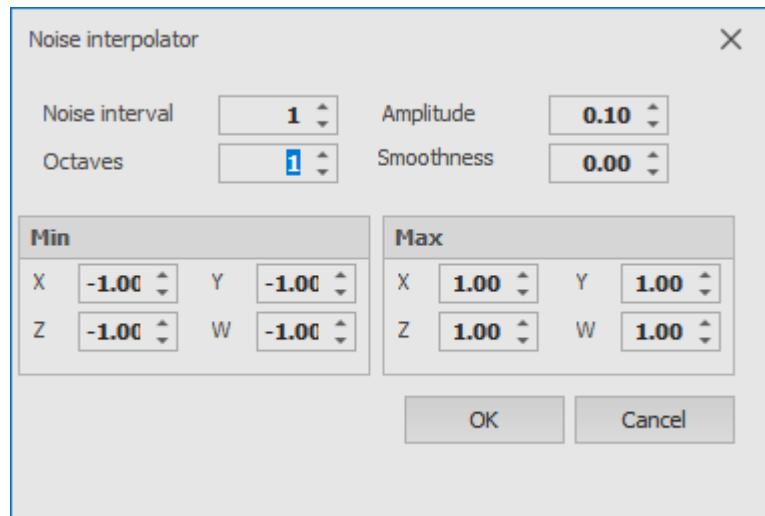


This feature allows for the creation of curve-based animations by smoothly transitioning between key frames using an adjustable spline curve. The trajectory between control/anchor points is controlled by tangent handles. Users can adjust the Ease-In and Ease-Out of a track by modifying a value between 0 and 1. Increasing the value towards 1 pushes the animation trajectory away from the current key frame towards the next, resulting in an Ease-In effect towards the subsequent frame.

In the Key Details section, users can manipulate the positions of the trajectory handles using X, Y, and Z values for the Left or Right handle per key frame. The initial key frame only has a Right Handle, while the ending key frame only has a Left Handle.

For key frames created between the first and last key frames, both the Left and Right trajectory handles can be modified. Alternatively, users can directly manipulate the Key frame handles in the viewport by selecting the respective object and the key frame in its trajectory.

- **Wire:** It is assigned to tracks which are wired with another track or a variable. When a user selects Wire Interpolator for a track, it opens the Wire Dialog Box to wire the respective track to a variable or track of another object matching its data type. for e.g. Flat Track can only be wired with a Float track of another object or a Float type variable. [See Wire Dialog](#)
- Noise:

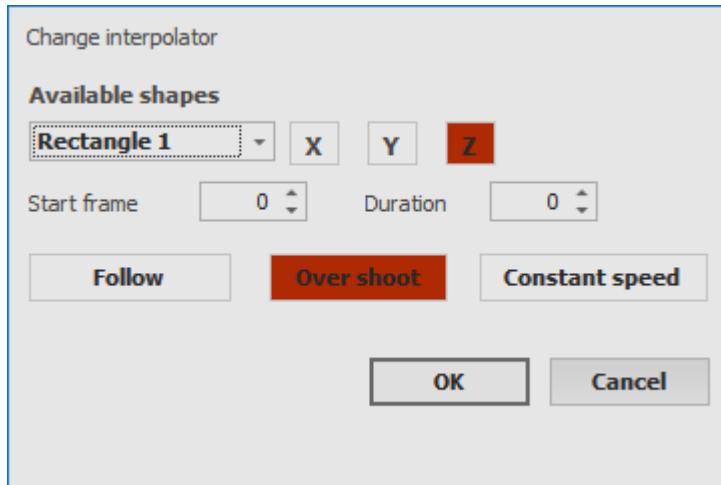


The Noise Interpolator generates random, fractal-based animation across a range of frames. This Interpolator applies noise controllers that are adjustable based on parameters. When the Noise Track is applied, it opens the Noise Interpolator Dialog Box, allowing for further customization and control over the noise-based animation effects.

- Boolean: Boolean interpolation assigns set values to key frames without any intermediate animation. The assigned value is instantly applied when the track reaches the respective key frame, resulting in immediate changes without gradual transitions.

- Numeric: The Numeric Interpolator is the default interpolation method applied to a Numeric Track of a Text object. This Interpolator is designed for counter-type animations, allowing for smooth transitions and interpolation of numeric values over time. It enables gradual changes from one value to another in a controlled manner. By utilizing the Numeric Interpolator, designers can create dynamic numerical animations such as counters or timers.

- Path:

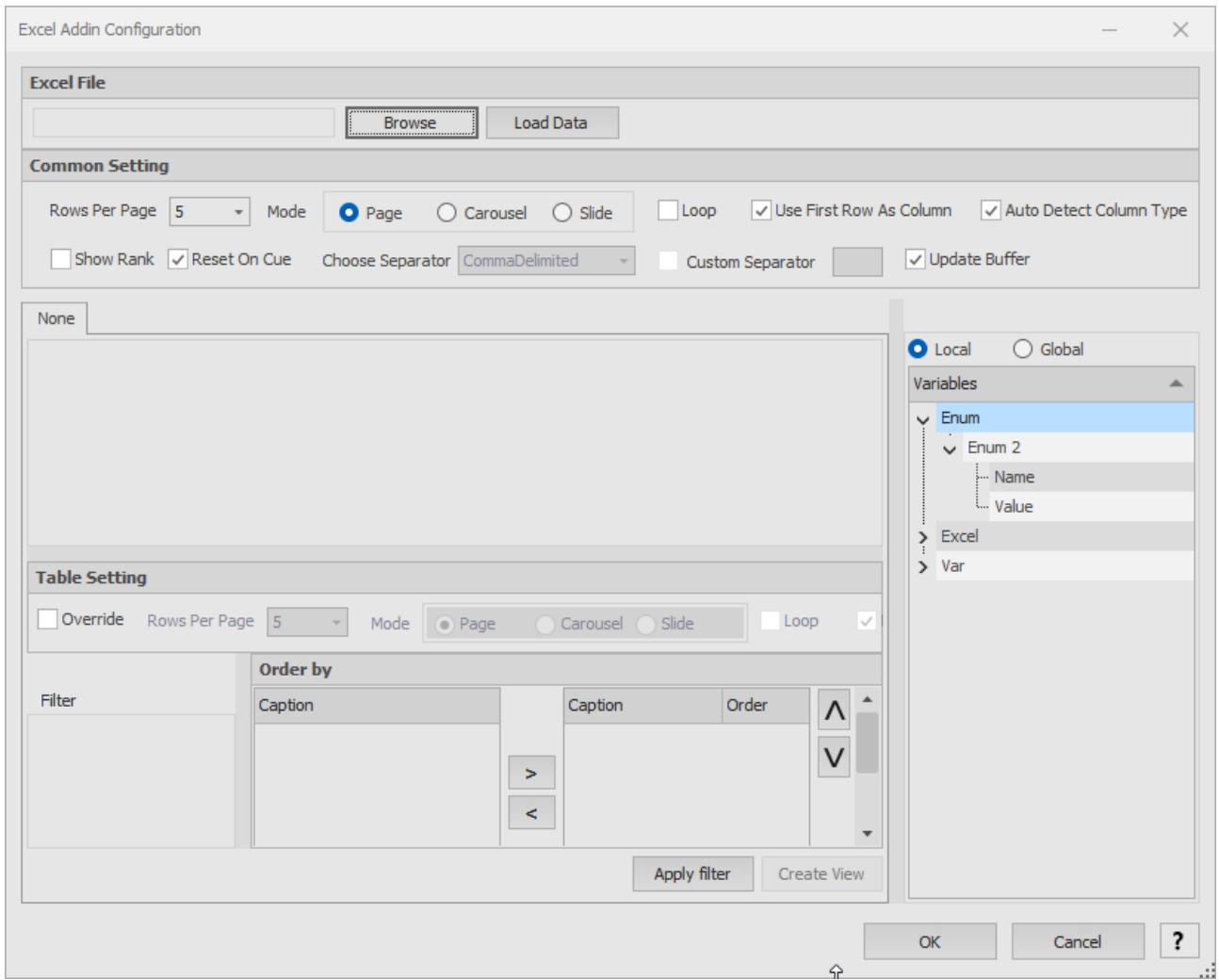


The Path Interpolator is a feature used to move an object along a predefined path based on the specified Path Interpolator Settings. This interpolation method enables precise translation of the object along the path, following the defined trajectory with accuracy and control.

3. User Tag: It can be assigned to key frames for dynamic run-time update of any object's track (transformation/parametric) in a scene.
4. Add Key (button): The "Add Key" button allows users to create a key frame on the selected track. This key frame is generated based on the values obtained after pressing the "Acquire Interpolated Value" button. As a result, there is no need to activate the Auto Key feature, and users can easily generate key frames on the selected track by utilizing modified values from the corresponding Transformation or Parametric track.
5. Acquire interpolated Value (button): When clicked, the Acquire interpolated Value button acquires the values of the selected track of the object and stores them in memory. These values are then added to the key frame when the Add Key button is pressed.

Excel Addin

It uses an Excel file as a source and reads data in the same tabular format as in Excel file and enables the user to link the graphic data. It supports connection data with CSV as well as TXT files.



- Browse: search for the file and add from any location on local system or network systems.
- Load Data: Loads the data from the selected file. The data is automatically loaded when browsed, but user can press this button to refresh the data from the file as well.
- Common Settings: These settings are applied on all the table sheets in the selected file. To override the common settings user can specify settings per table using the table settings.
 - Row per page: select number of rows of data to be extracted for single display of data. The Row per page is set according to the number of items to be displayed in one data set of graphic. When user wants to display a team sheet of soccer for 11 players, the row per page would be set to 11. and when showing data for a Map where all constituencies are required to be colored, the All option is selected for it.
 - Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.
 - Page: For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.
 - Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the

change is like this because the 1st item will exit and will become the 6th item, similarly, the 2nd item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.

- Slide: This mode is designed as another menu mode to accommodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required, but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.
- Use first row as column: When enabled, sets first row of the file as column titles and removes the 1st row from data. If disabled, sets first row of the file as column titles and keeps the 1st row in data
- Auto detect column type: Detects the data-type of the columns based on the data in the first Row of the excel. User can manually define the column types by selecting the required type using the drop down arrow in each column of the table data shown in the interface.
- Loop: Repeats the data once reached end of the records. The end of records is calculated based on the selected mode.
- Maintain Schema: Locks the data assignment with the column. This is helpful incase rows are added inbetween the existing data set after the data is mapped to the graphic.
- Show Rank: Adds a Rank column in the existing data set. This is useful to see the row numbers and to filter the data based on row number if there no similar column in the data.
- Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
- Choose Separator: The parameters gets enabled when a Text file is selected for data import. User can select from Comma, Semi-colon or Tab Delimited option being the commonly used setting. Enable custom separator to specify in the attached text box if another separator is used.
- Custom Separator: When enabled, Allows user to specify a custom data separator. This is helpful when a special character or a glyph is specified as a data separator in the text file.
- Update Buffer: When enabled, the data updates are fetched automatically in case of changes in the sourced data file. when unchecked, the update is fetched only when the Update Buffer action is called for this addin in the scene. The case is useful when the data is wired, but only requires to be updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.
- Data: Displays tabs based on data. There can be multiple tabs of data depending on the fetched data. It is displayed in form of columns. It displays all the columns of the table. User can change the column type using the Arrow drop down displaying different type of data types. The data is displayed as per filter and settings applied in the common settings.
- Table Setting: The parameters in Table setting are similar to what is available in the common settings. They are used to override the common settings for the respective table data. Please see the parameters defined in the Common parameters for more info.
- Variables: Displays list of variables in the scene which can be dragged to the filter column to filter the data as per the data point in variable.
 - Local: gives the list of local variables which are only available in the scene in which it is created
 - Global: gives the list of global variables which are available/accessible in any scene

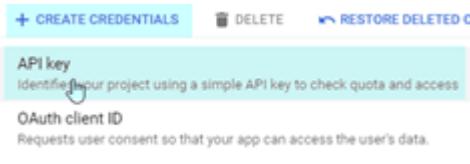
- Filter: Filters the data table as per the defined filter. Click on the + button to add filter.
 - To filter table data with static or variable value adding conditions like “begins with”, “ends with”, “equals to”
 - Order by: re-arranges the content of the sheets in the order specified.
- Apply Filter: Applies the selected settings to the Data Table(s).
- Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify different filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

Google Sheet Addin

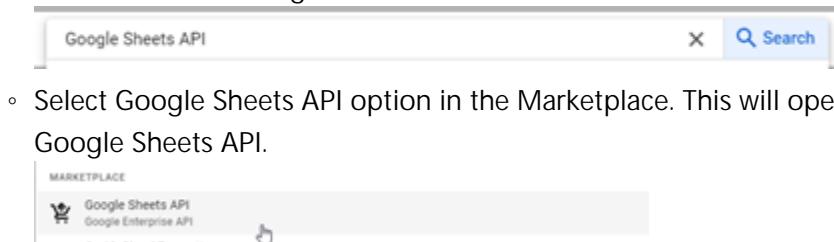
Google Sheets Addin: A data source adds in that fetches data from Google Sheets. As the sheet is stored in Cloud, It has an ID and not a local disk path. The API Key of the User would provide the data access to get the data from the Google Sheets stored in their ID and Sheet ID would provide the location.

- API Key - This is required once per user. One API can call for multiple sheets shared by the user.
- Sheet ID - Would change as per sheet from which the user wants to connect data. Follow the Below steps to get data from Google Sheets.

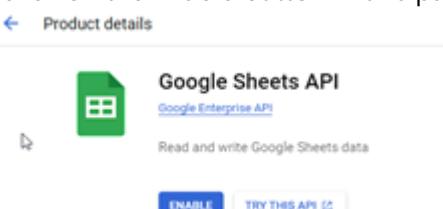
- Users need to create a Google Sheet API key from the below link. <https://console.cloud.google.com/apis/credentials>
- Click on Create Credentials Button in the and select API Key.



- Once Clicked, it will create an API Key and add an entry in the API Keys table.
- Click on the 'SHOW KEY' Button to reveal the API Key and paste it in notepad or any location.
- Next, Search for Google Sheets API in the search bar.



- Click on the Enable button in this page.



- Once the Google Sheets API is enabled, the user needs to locate the Sheet ID. To do that, open the Google Sheet in browser and click on the  "Share" Button to open the share Google Sheet Dialog Box.

- Select the "Anyone with the Link" option and as Wasp3D only requires view rights, you can choose to only select as Viewer, else the
- From the copied string, select the Sheet ID which would be the middle part of the complete http string as highlighted in the image below:

<https://docs.google.com/spreadsheets/d/1tjBBEQtML1B62AUGoAdyCQPnpm1xMX71B62AUGoAdyA/edit?usp=sharing>

- Copy the Sheet ID through copy link and replace with the Sheet ID Text in the below link. Similarly, paste the copied API key and replace it with the API Key Text.

<https://sheets.googleapis.com/v4/spreadsheets/Sheet ID/values/Sheet1?key=API Key>

Note: Colors of SheetID and API Key are only to highlight.

- When the link is complete and looks something like this:

<https://sheets.googleapis.com/v4/spreadsheets/>

1gmXuusHqODI5ogdOFsQyNtXzDFeCE1gmXuusHqODI5ogGtU/values/Master%20List?key=AlzaSyurousHqODI5ogdYEiQH_di4_w4zr_TqxWmE

- Paste it in the Enter URL Text Box in the Google Sheets Addin.

- Enter URL: Enter an http address.

- Load Data: loads and reflects the data in the instance.

- Common Settings: These settings are applied on all the table sheets in the selected file. To override the common settings user can specify settings per table using the table settings.

- Row per page: select number of rows of data to be extracted for single display of data. The Row per page is set according to the number of items to be displayed in one data set of graphic. When user wants to display a team sheet of soccer for 11 players, the row per page would be set to 11. and when showing data for a Map where all constituencies are required to be colored, the All option is selected for it.

- Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.

- Page: For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.

- Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the change is like this because the 1st item will exit and will become the 6th item, similarly, the send item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.

- Slide: This mode is designed as another menu mode to accommodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required, but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.

- Loop: Repeats the data once reached end of the JSON.
- Requery (in sec): Sets the time frame to re-check for data updates in the selected file.
- Show Rank: Adds a Rank column in the existing data set. This is useful to see the row numbers and to filter the data based on row number if there no similar column in the data.
- Maintain Schema: Locks the data assignment with the column. This is helpful in case rows are added in between the existing data set after the data is mapped to the graphic.
- Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
- Update Buffer: When enabled, the data updates are fetched automatically in case of changes in the sourced data file. when unchecked, the update is fetched only when the Update Buffer action is called for this Addin in the scene. The case is useful when the data is wired, but only requires to be updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.
- **Data:** Is displayed in form of columns. It displays all the columns of the table. User can change the column type using the Arrow drop down displaying different type of data types. The data is displayed as per filter and settings applied in the common settings.
- Table Setting: The parameters in Table setting are similar to what is available in the common settings. They are used to override the common settings for the respective table data. Please see the parameters defined in the Common parameters for more info.
- Variables: Displays list of variables in the scene which can be dragged to the filter column to filter the data as per the data point in variable.
 - Local: gives the list of local variables which are only available in the scene in which it is created
 - Global: gives the list of global variables which are available/accessible in any scene
- Filter: Filters the data table as per the defined filter. Click on the + button to add filter.
 - To filter table data with static or variable value adding conditions like "begins with", "ends with", "equals to"
 - Order by: re-arranges the content of the sheets in the order specified.
- Apply Filter: Applies the selected settings to the Data Table(s).
- Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify different filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

HemiCycle Addin

The HemiCycle Addin provides positional and visual data for representing grouped or distributed datasets—such as election results—in a radial or hemicycle (semi-circular) layout. It enables designers to visualize seat allocations by party or category using customizable layout parameters and data sources.

The output generated by the addin is not visual on its own. It must be mapped to a Particle Object, which consumes the dataset and renders the actual seat visualization.

Functionality Overview

- Supports manual data entry and live data source binding
- Generates per-seat data with position, color, label, and rank (optional)
- Provides support for both radial and hemicycle layouts
- Seamlessly integrates with Particle Object through particle data binding

Parameters

Parameter	Description
Arrangement	Choose layout style: Hemicycle (semi-circle) or Radial (full circle)
No. of Parties	Total number of distinct entries (e.g., parties)
Total Seats	Total number of seats to be arranged
Height Offset	Vertical elevation applied to the layout, useful for depth layering
Angle	Applicable to Hemicycle only; defines the span of the arc
Rows	Number of concentric rows used for seat arrangement
Radius	Base radius for the outermost row
Radial Offset	Distance between each row toward the center
Update Buffer	Enables or disables real-time dataset updates (see below)
Show Rank	Adds a numerical rank column to each seat row (see below)

Update Buffer

Controls how and when the internal dataset updates:

- Enabled: Automatically updates data table and linked visuals in real-time whenever the input data changes.
- Disabled: The dataset is static until explicitly refreshed by the user using the Update Buffer Action.

The Update Buffer Action can be triggered via:

- Timeline
- Action Set
- Timer Event
- Any other event logic in the scene

Use this for synchronized transitions or when updates must be timed precisely.

Show Rank : When enabled, adds a Rank column as the first column in the dataset output.

- Rank values begin at `1` and go up to `Total Seats`.
- Useful for sequential animations, seat numbering, or indexing purposes.

Data Input Methods

Manual Input

Allows the user to enter data directly into a table interface.

Column	Description
Party	Name or identifier of the group (e.g., political party)
Seats	Number of seats won/assigned to that group

Column	Description
Color	Color assigned to the seats of that group (hex or RGB)
Label	Optional label or abbreviation (e.g., "BJP", "INC")

After data entry, the addin generates one row per seat with calculated position, color, and label based on layout settings.

Data Source Binding : When using an external data source (e.g., XML, JSON, CSV):

- User must map relevant columns from the source to the addin fields:
 - Party
 - Seats
 - Color
 - Label

Once mapped, data from the source is continuously monitored and automatically updated (if Update Buffer is enabled).

Output Dataset Structure

The addin produces a structured table with the following columns:

Column	Description
(Rank)	Optional rank value (if Show Rank is enabled)
Party	Party or group name
Seats	Number of seats assigned to this party
Color	Display color of the seat
Label	Abbreviation or identifier
Position	X/Y (or radial) coordinates for each seat in layout

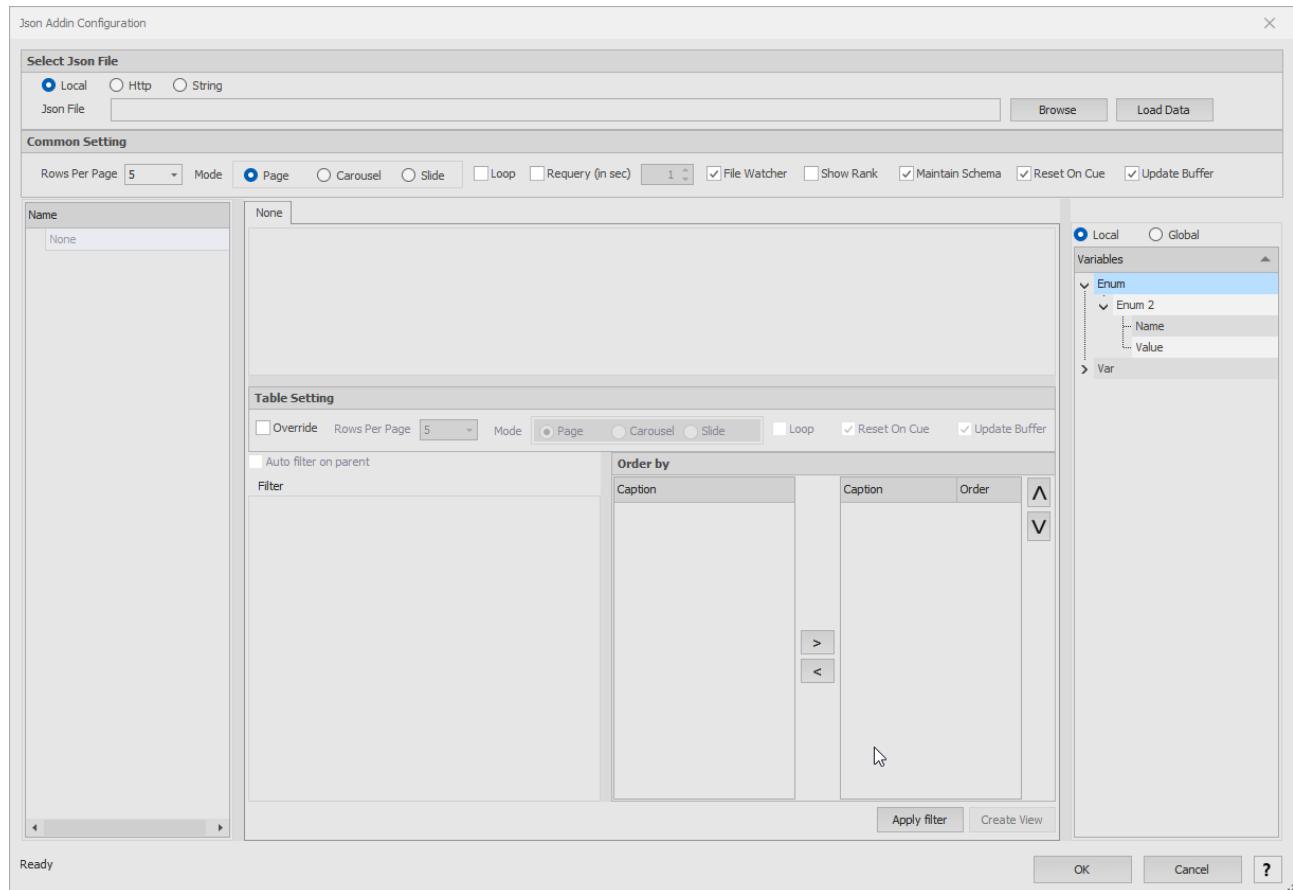
Visual Representation : The data generated by the HemiCycle Addin must be applied to a Particle Object:

- The Particle Data parameters of the object are mapped to:
- Position → X/Y layout position of each seat
- Color → Seat fill color
- Label → Seat identifier
- Rank → Numerical identifier (optional)

This allows full control over seat appearance, effects, animations, and behavior through the particle system.

Json Addin

A data source add in that fetches data from a JSON file in a tabular format reading the nodes and their attributes. User can use both locally & network stored (e.g., JSON) JSON files as a source.



- Select JSON file:
 - Local: Option to select a locally saved JSON file
 - HTTP: Option to enter an http address.
 - String: One can create the JSON in the given drop down space.
 - Browse: locate the file and add from the local system.
 - Load Data: loads and reflects the data in the instance.
- Common Settings: These settings are applied on all the table sheets in the selected file. To override the common settings user can specify settings per table using the table settings.
 - Row per page: select number of rows of data to be extracted for single display of data. The Row per page is set according to the number of items to be displayed in one data set of graphic. When user wants to display a team sheet of soccer for 11 players, the row per page would be set to 11. and when showing data for a Map where all constituencies are required to be colored, the All option is selected for it.
 - Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.
 - Page: For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.

- Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the change is like this because the 1st item will exit and will become the 6th item, similarly, the 2nd item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.
 - Slide: This mode is designed as another menu mode to accommodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required, but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.
 - Loop: Repeats the data once reached end of the JSON.
- Requery (in sec): Sets the time frame to re-check for data updates in the selected file.
- File watcher: Automatically loads the selected file once the changes are done and saved.
- Show Rank: Adds a Rank column in the existing data set. This is useful to see the row numbers and to filter the data based on row number if there no similar column in the data.
- Maintain Schema: Locks the data assignment with the column. This is helpful in-case rows are added in between the existing data set after the data is mapped to the graphic.
- Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
- Update Buffer: When enabled, the data updates are fetched automatically in case of changes in the sourced data file. when unchecked, the update is fetched only when the Update Buffer action is called for this Addin in the scene. The case is useful when the data is wired, but only requires to be updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.
- Tree Structure: The tree structure of the JSON is displayed in this window which displays the data hierarchy of the selected JSON file.
- Data: Displays tabs based on data. There can be multiple tabs of data depending on the fetched data. It is displayed in form of columns. It displays all the columns of the table. User can change the column type using the Arrow drop down displaying different type of data types. The data is displayed as per filter and settings applied in the common settings.
- Table Setting: The parameters in Table setting are similar to what is available in the common settings. They are used to override the common settings for the respective table data. Please see the parameters defined in the Common parameters for more info.
- Variables: Displays list of variables in the scene which can be dragged to the filter column to filter the data as per the data point in variable.
 - Local: gives the list of local variables which are only available in the scene in which it is created
 - Global: gives the list of global variables which are available/accessible in any scene
- Filter: Filters the data table as per the defined filter. Click on the + button to add filter.

- To filter table data with static or variable value adding conditions like "begins with", "ends with", "equals to"
- Order by: re-arranges the content of the sheets in the order specified.
- Apply Filter: Applies the selected settings to the Data Table(s).
- Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify different filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

ODBC Addin

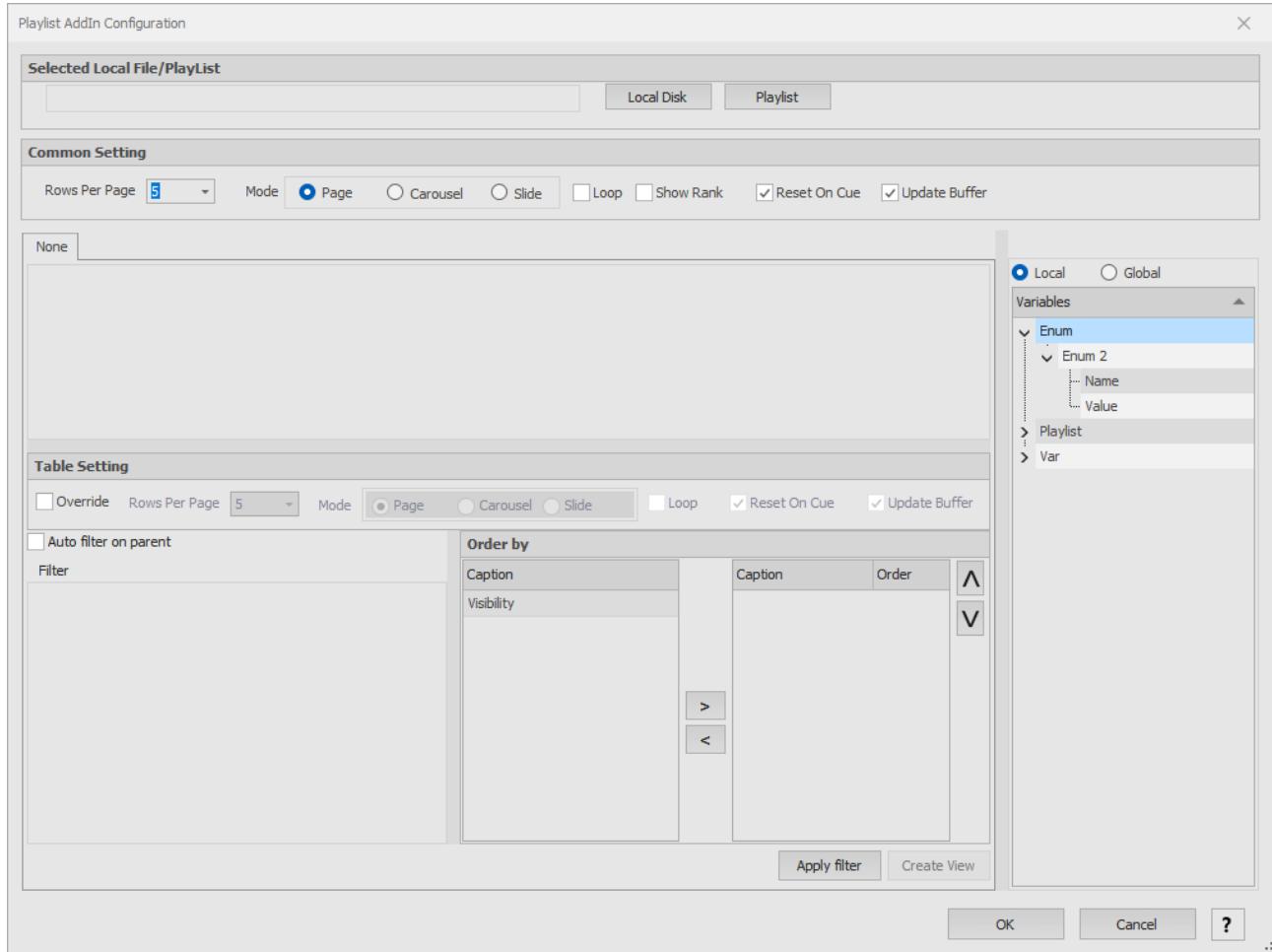
Addin that reads data from ODBC data sources available on the network or respective system. User should configure the ODBC data source and install the relevant drivers using the 'Set up data sources (ODBC)' utility in Administrative Tools available in Control Panel in Microsoft Windows.

- Connection details: Displays the parameters required to connect to a Database server to get the data into the template.
 - Connections: Creates a new or gives an option to select the saved connections data.
 - DSN Type: Choose between System or File type DSN whichever gateway is created or to be added as new source to access the ODBC file or server.
 - Data Sources: Displays the list of available data sources for ODBC connection User can click on the + button to open the ODBC Datasource Administration window to add any other data source.
 - DSN Name: Enter or Select the DSN name to be connect the data source.
 - User: Sets the User Name to access the DSN.
 - Password: Sets the Password to access the DSN.
 - Connect: The connect button is enabled when the Connection parameters are entered and user is ready to load the data from the selected database.
 - Save Connection: When pressed, saves the existing setting as a connection in the Addin configuration. The saved connections are displayed in the Connection ComboBox list.
- Procedure or Query: Allows user to select a stored procedure or write a Database query to get the data from the connected database.
 - Stored Procedure: The stored procedure from the connected database are displayed in the Stored Procedure Combobox.
 - Custom Statements: Select the Custom Statements radiobox to write a DB query to get the data from the connected database.
- Params: Data Parameters are displayed when either specified in the Stored Procedure or the query. User can declare variable parameters using the \$variable name format.
 - Get Query Param: Extract the query if mentioned in the custom statement. This enables the option of wiring this query to a variable or manually entering the data to filter data fields.
 - Execute: Executes the query with the parameters (when specified) and loads the data in the data table.
- Common Settings:
 - Data Refresh: no. of seconds in which the data is refreshed.

- Row per page: select number of rows of data to be extracted for single display of data.
 - Loop: Repeats the data once reached end of the Table.
 - Maintain Schema: Locks the data assignment with the column. This is helpful incase rows are added inbetween the existing data set after the data is mapped to the graphic.
 - Show Rank: Adds a Rank column in the existing data set. This is useful to see the row numbers and to filter the data based on row number if there no similar column in the data.
 - Update Buffer: When enabled, the data updates are fetched in page and updated on-air. When unchecked, the update is fetched but the text is not updated on-air. The data is updated on-air, when the update buffer action is called using an action set or at required frame on the timeline. The case is useful when the data is wired, but only requires to be updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.
 - Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
- Data: Displays tabs based on data. There can be multiple tabs of data depending on the fetched data. It is displayed in form of columns. It displays all the columns of the table. User can change the column type using the Arrow drop down displaying different type of data types. The data is displayed as per filter and settings applied in the common settings.
- Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.
 - Page: For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.
 - Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the change is like this because the 1st item will exit and will become the 6th item, similarly, the 2nd item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.
 - Slide: This mode is designed as another menu mode to accomodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required, but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.
- Filter: Filters the data table as per the defined filter. Click on the + button to add filter.
 - To filter table data with static or variable value adding conditions like "begins with", "ends with", "equals to"
 - Order by: re-arranges the content of the sheets in the order specified.
- Apply Filter: Applies the selected settings to the Data Table(s).
- Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify differnt filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

Playlist Addin

Provides the list of scenes available in a folder or list of instances available in a playlist made using DataBuzz or Sting Client. These scenes can be wired using the WScene variable to Load, Play, Stop or Unload using the Waspi Actions in the Action Control Window.



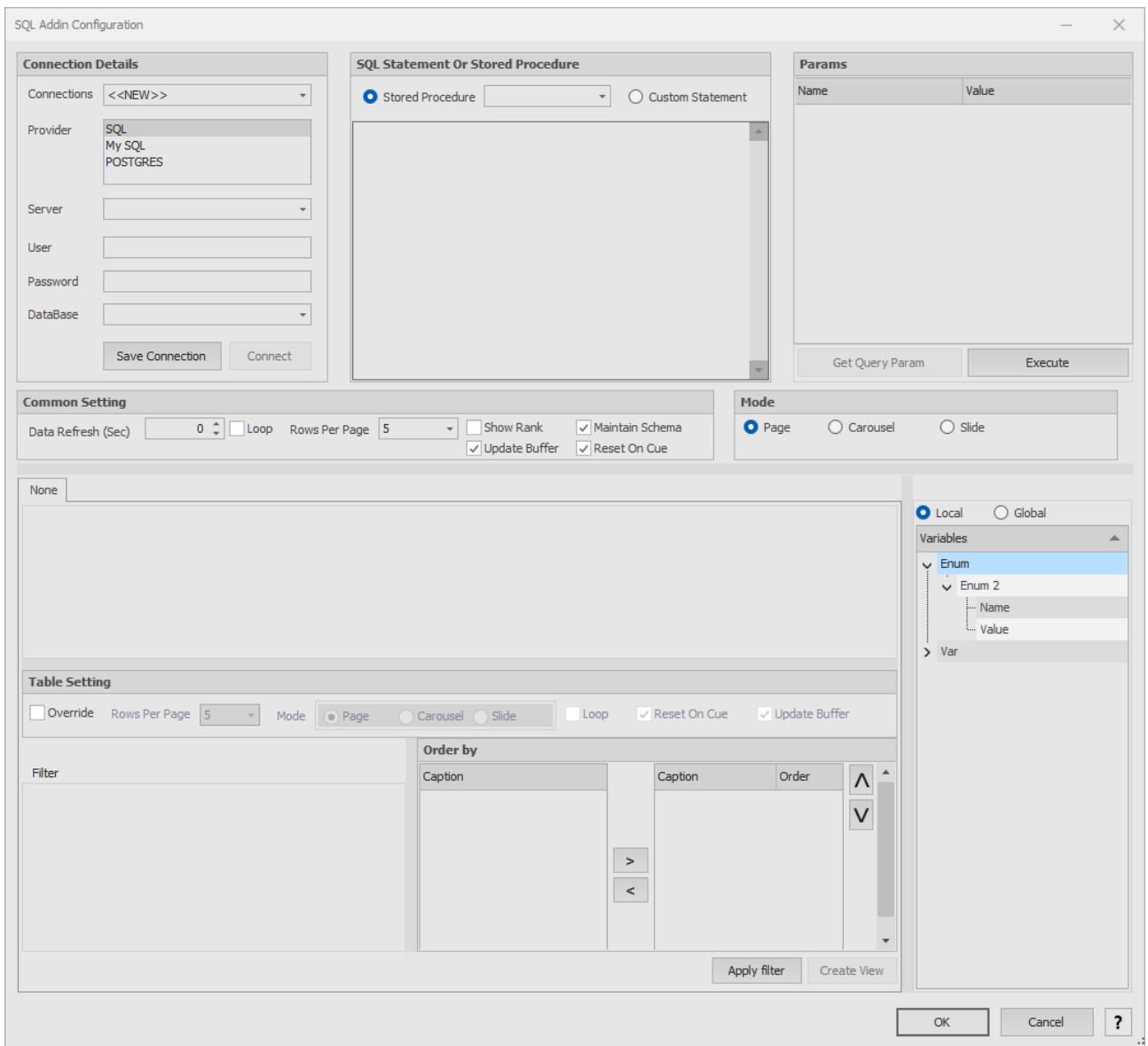
- Select Local File/Playlist: Allow user to get the templates from a folder or a playlist.
 - Local Disk: Opens select folder dialog box to select the folder from which all the templates will be added into the table.
 - Playlist: Opens the playlist pool box to select the playlist from which all the templates will be added into the table.
- Common Settings:
 - Data Refresh: no. of seconds in which the data is refreshed.
 - Row per page: select number of rows of data to be extracted for single display of data.
 - Loop: Repeats the data once reached end of the Table.
 - Show Rank: Adds a Rank column in the existing data set. This is useful to see the row numbers and to filter the data based on row number if there no similar column in the data.
 - Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
 - Update Buffer: When enabled, the data updates are fetched in page and updated on-air. When unchecked, the update is fetched but the text is not updated on-air. The data is updated on-air, when the update buffer action is called using an action set or at required frame on the timeline. The case is useful when the data is wired, but only requires to be

updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.

- Data: Displays tabs based on data. There can be multiple tabs of data depending on the fetched data. It is displayed in form of columns. It displays all the columns of the table. User can change the column type using the Arrow drop down displaying different type of data types. The data is displayed as per filter and settings applied in the common settings.
- Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.
 - Page:.. For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.
 - Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the change is like this because the 1st item will exit and will become the 6th item, similarly, the 2nd item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.
 - Slide: This mode is designed as another menu mode to accommodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required, but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.
- Filter: Filters the data table as per the defined filter. Click on the + button to add filter.
 - To filter table data with static or variable value adding conditions like "begins with", "ends with", "equals to"
 - Order by: re-arranges the content of the sheets in the order specified.
- Apply Filter: Applies the selected settings to the Data Table(s).
- Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify different filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

SQL Addin

A data-source addin that reads data from a Database in the network, either using Stored Procedure or custom SQL statement. It supports data import from Microsoft SQL, MySQL and PostgreSQL.



- Connection details: Displays the parameters required to connect to a Database server to get the data into the template.
 - Connections: Creates a new or gives an option to select the saved connections data.
 - Provider: Select the data structure to import the data from the respective Database Provider.
 - Server: Display the list of database servers in the local machine or across the network.
 - User: Sets the username to access the database on the selected server.
 - Password: Sets the password for the username to access the database on the selected server.
 - Database: Sets the database after correct username and password are entered in the username and password field for the selected server.
 - Save Connection: When pressed, saves the existing setting as a connection in the Addin configuration. The saved connections are displayed in the Connection ComboBox list.
 - Connect: The connect button is enabled when the Connection parameters are entered and user is ready to load the data from the selected database.
- SQL statement or stored procedure: Allows user to select a stored procedure or write a SQL query to get the data from the connected database.

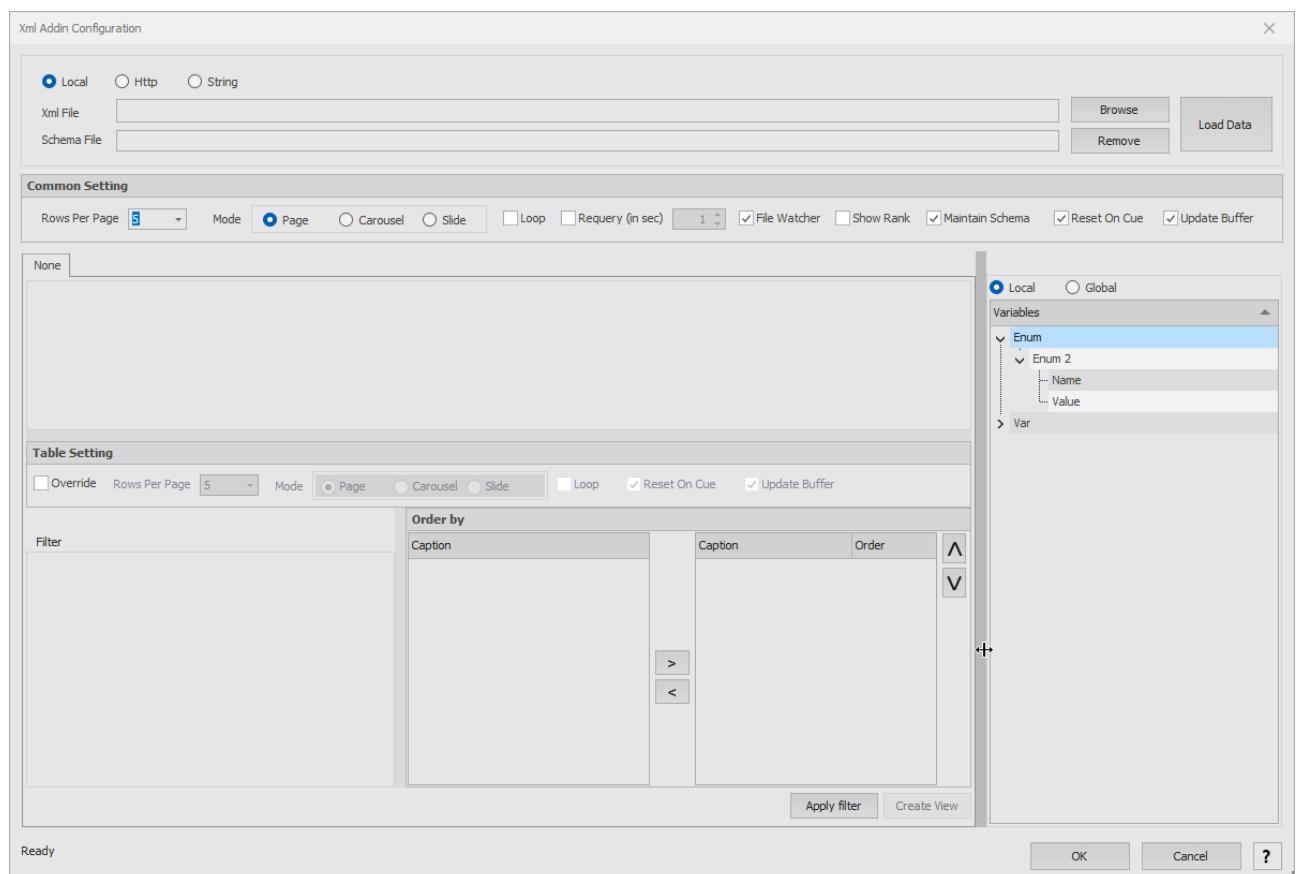
- Stored Procedure: The stored procedure from the connected database are displayed in the Stored Procedure Combobox.
 - Custom Statements: Select the Custom Statements radiobox to write a SQL query to get the data from the connected database.
- Params: Data Parameters are displayed when either specified in the Stored Procedure or the SQL statement. User can declare variable parameters using the \$variable name format.
 - Get Query Param: Extract the query if mentioned in the custom statement. This enables the option of wiring this query to a variable or manually entering the data to filter data fields.
 - Execute: Executes the query with the parameters (when specified) and loads the data in the data table.
- Common Settings:
 - Data Refresh: no. of seconds in which the data is refreshed.
 - Row per page: select number of rows of data to be extracted for single display of data.
 - Loop: Repeats the data once reached end of the SQL Table.
 - Maintain Schema: Locks the data assignment with the column. This is helpful incase rows are added inbetween the existing data set after the data is mapped to the graphic.
 - Show Rank: Adds a Rank column in the existing data set. This is useful to see the row numbers and to filter the data based on row number if there no similar column in the data.
 - Update Buffer: When enabled, the data updates are fetched in page and updated on-air. When unchecked, the update is fetched but the text is not updated on-air. The data is updated on-air, when the update buffer action is called using an action set or at required frame on the timeline. The case is useful when the data is wired, but only requires to be updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.
 - Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
- Data: Displays tabs based on data. There can be multiple tabs of data depending on the fetched data. It is displayed in form of columns. It displays all the columns of the table. User can change the column type using the Arrow drop down displaying different type of data types. The data is displayed as per filter and settings applied in the common settings.
- Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.
 - Page: For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.
 - Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the change is like this because the 1st item will exit and will become the 6th item, similarly, the 5th item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.
 - Slide: This mode is designed as another menu mode to accomodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required,

but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.

- Filter: Filters the data table as per the defined filter. Click on the + button to add filter.
 - To filter table data with static or variable value adding conditions like "begins with", "ends with", "equals to"
 - Order by: re-arranges the content of the sheets in the order specified.
- Apply Filter: Applies the selected settings to the Data Table(s).
- Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify different filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

XML Addin

A data source add in that fetches data from a xml file in a tabular format reading the nodes and their attributes. User can use both locally & network stored (e.g., RSS feed) xml files as a source.



- Local: Option to select a locally saved XML file
- HTTP: Option to enter an http address.
- String: One can create the XML in the given drop down space.
- XML file: displays the browsed XML file with complete path.
- Browse: locate the file and add from the local system.

- Schema File: loads the schema file with the local XML if present. Schema can be defined to read the selected XML file.
- Load Data: loads and reflects the data in the instance.
- Common Settings: These settings are applied on all the table sheets in the selected file. To override the common settings user can specify settings per table using the table settings.
 - Row per page: select number of rows of data to be extracted for single display of data. The Row per page is set according to the number of items to be displayed in one data set of graphic. When user wants to display a team sheet of soccer for 11 players, the row per page would be set to 11. and when showing data for a Map where all constituencies are required to be colored, the All option is selected for it.
 - Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.
 - Page: For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.
 - Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the change is like this because the 1st item will exit and will become the 6th item, similarly, the 2nd item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.
 - Slide: This mode is designed as another menu mode to accommodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required, but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.
 - Loop: Repeats the data once reached end of the XML.
 - Requery (in sec): Sets the time frame to re-check for data updates in the selected file.
 - File watcher: Automatically loads the selected file once the changes are done and saved.
 - Show Rank: Adds a Rank column in the existing data set. This is useful to see the row numbers and to filter the data based on row number if there no similar column in the data.
 - Maintain Schema: Locks the data assignment with the column. This is helpful incase rows are added inbetween the existing data set after the data is mapped to the graphic.
 - Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
 - Update Buffer: When enabled, the data updates are fetched automatically in case of changes in the sourced data file. when unchecked, the update is fetched only when the Update Buffer action is called for this addin in the scene. The case is useful when the data is wired, but only requires to be updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.
- Data: Displays tabs based on data. There can be multiple tabs of data depending on the fetched data. It is displayed in form of columns. It displays all the columns of the table. User can change

the column type using the Arrow drop down displaying different type of data types. The data is displayed as per filter and settings applied in the common settings.

- Table Setting: The parameters in Table setting are similar to what is available in the common settings. They are used to override the common settings for the respective table data. Please see the parameters defined in the Common parameters for more info.
- Variables: Displays list of variables in the scene which can be dragged to the filter column to filter the data as per the data point in variable.
 - Local: gives the list of local variables which are only available in the scene in which it is created
 - Global: gives the list of global variables which are available/accessible in any scene
- Filter: Filters the data table as per the defined filter. Click on the + button to add filter.
 - To filter table data with static or variable value adding conditions like "begins with", "ends with", "equals to"
 - Order by: re-arranges the content of the sheets in the order specified.
- Apply Filter: Applies the selected settings to the Data Table(s).
- Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify different filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

Chroma

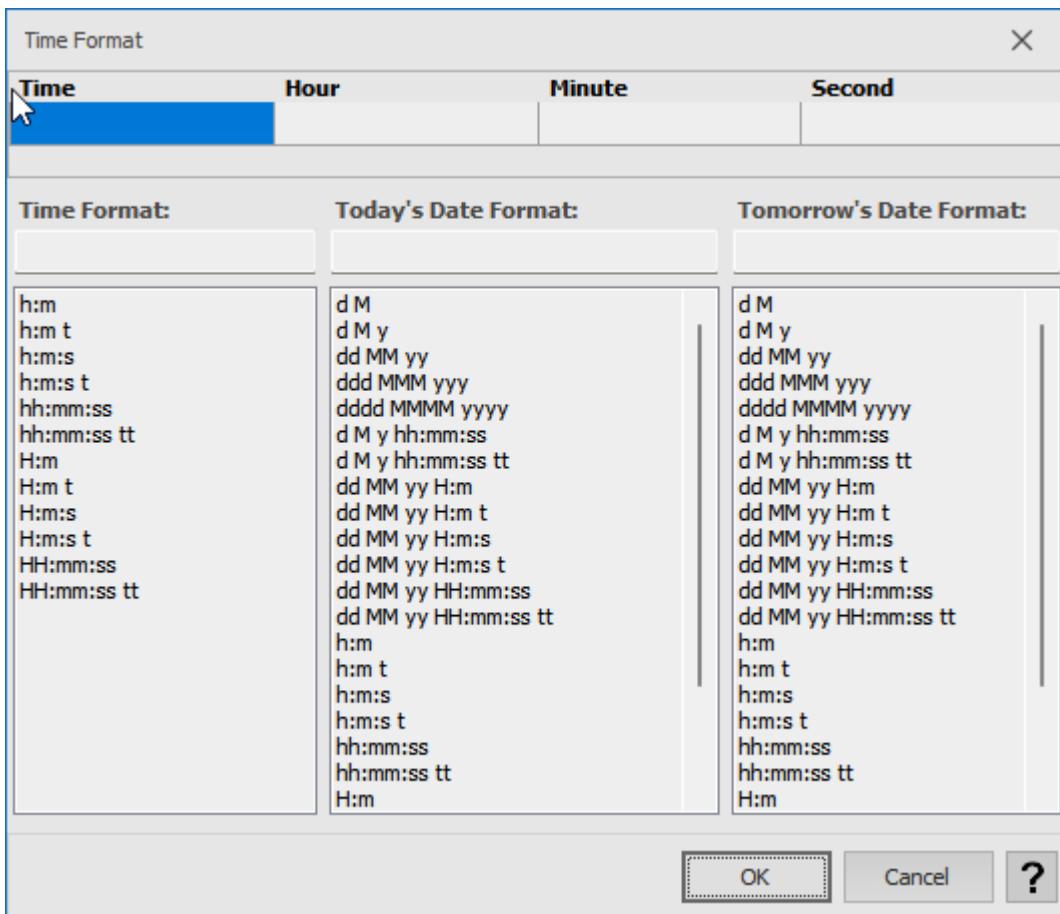
Chroma Parameters

1. Camera – Select the Camera to which the Chroma settings are to be done. The camera selection is based on the Input which is received from the camera. This way user can assign different chroma values to each camera view as the close-shot and long shot may have different chroma values.
2. Live – Click on the Live button toggle between Live Gamma and Live video output. When Live is selected, it displays the original live input feed. When Live (Gamma) is selected, it displays the pre-processed live input with the assigned Gamma value.
3. Composite – displays the composite output after the Chroma settings are applied on the original video.
4. Live Matte – Shows the black and white image of the Chroma area. Black is the keyed-out area and white is the trimmed image.
5. Scene Matte – Displays the matte of the Graphic Scene(s) loaded in the Engine.
6. Composite Matte – Displays a composite of Live and Scene matte.
7. Invert Matte – Is used to display graphics in the Chroma Area. It is helpful where user wants to display graphics in the Football Field but below the players or in Water below the boat.
8. Key Color – It assigns a range between the 2 colors selected which are to be keyed. User can select 2 colors if there are different green shades due to lighting or difference in the color in the chroma studio.
9. Exclude Colors – it sets the color range which is to be excluded from the Chroma Color. There can be near hues of the selected chroma colors which should be excluded from the Chroma Color. User can specify the colors in the Exclude color values. Up to 2 colors can be defined for Exclude color.

10. Gamma – Gamma controls the luminance of colors in the video. A value of 100 is equal to the original live input. Higher Gamma makes the video darker, and a lower value increases the brightness of colors.
11. Density – modifies the intensity of the key color range. Increase power to remove more color variation/range of the selected key color range.
12. Matte – The Matte Black and Matte White counter defines a range to adjust the removal of the Chroma BG. This helps to define a dark to light range of the key color which will get removed from the video signal.
13. Threshold – Defines a cut off value of the black level. Matte Blackness and whiteness will work on levels above this.
14. Spill Brightness - controls the luminance or brightness in areas affected by spill. This adjustment is particularly useful when the Spill Threshold level is increased to reduce the influence of spill color.
15. Spill Threshold - Adjusting the spill threshold allows for fine-tuning the chroma key effect to achieve the desired level of precision in separating the foreground subject from the background. A higher spill threshold might result in a more precise color removal, while a lower threshold could allow for an overlay of the chroma color to remain around the edges of the subject.
16. **Pre-Process** – Enables user to update to modify the color balance of the video. All the modifications are applied on the video before the chroma is processed. So, changing these settings is like modifying the original video.
 - Enable – Toggles the pre-chroma color balancing of the chroma video.
 - Hue – Allows user to select a color which the user can change or modify in the original live video. Double click on the Empty Color box to select the color from the video in the Render Engine or anywhere in the V-Set Pro interface.
 - Rotate – Moves the Hue value from 0–360 to change the selected color.
 - Tolerance – Adjusts the color range which would get affected with the selected color change. Increase the tolerance to increase the selected colors nearby range. The extended range is displayed using a highlighted rectangle between the color.
 - Brightness, Contrast and Saturation – Controls the Brightness, Contrast and Saturation of the Chroma input source. All the changes are applied on the chroma video before processing the chroma. Hence it also enhances the video to help and remove the chroma color from the video.
17. Post Process – Users can adjust the brightness, contrast, and saturation of the feed. You can use these post process controls to negotiate on the changes done in the pre-process settings as well as enhancing the video after the chroma keying is processed.

Clock

Clock enables the user to create customized digital and analog clock using data from system time and provide the value for Clock hand Angles. Addin also provide the system time in the user selected formats. This is a custom addin created using the Wasp3D add-in SDK demonstrating how users can create custom engine-addins to provide data to Wasp3D render engine. This addin can also be used in the design to create analog clocks. where the angle of each hand is provided in the addin table.



When a clock addin is added to the scene and the respective formats are selected, an addin table is created with the respective time format and showing days for Today, Tomorrow as per the defined time format.

This time is updated in addin in realtime.

Key features

1. Customized Digital and Analog Clocks: Users can create digital and analog clocks with custom designs and settings.
2. Clock Hand Angles: The add-in allows users to specify the angles for each clock hand (e.g., hour, minute, second hands) to create unique analog clock designs.
3. System Time Integration: The add-in uses the system time to display the current time on the clocks in real-time.
4. Time Formats: Users can select different time formats for the clocks, such as 12-hour or 24-hour formats.
5. Date Information: The add-in displays date information, including days for "Today" and "Tomorrow," based on the defined time format.
6. Real-Time Updates: The clock's time is updated in real-time, ensuring that it always reflects the current system time.

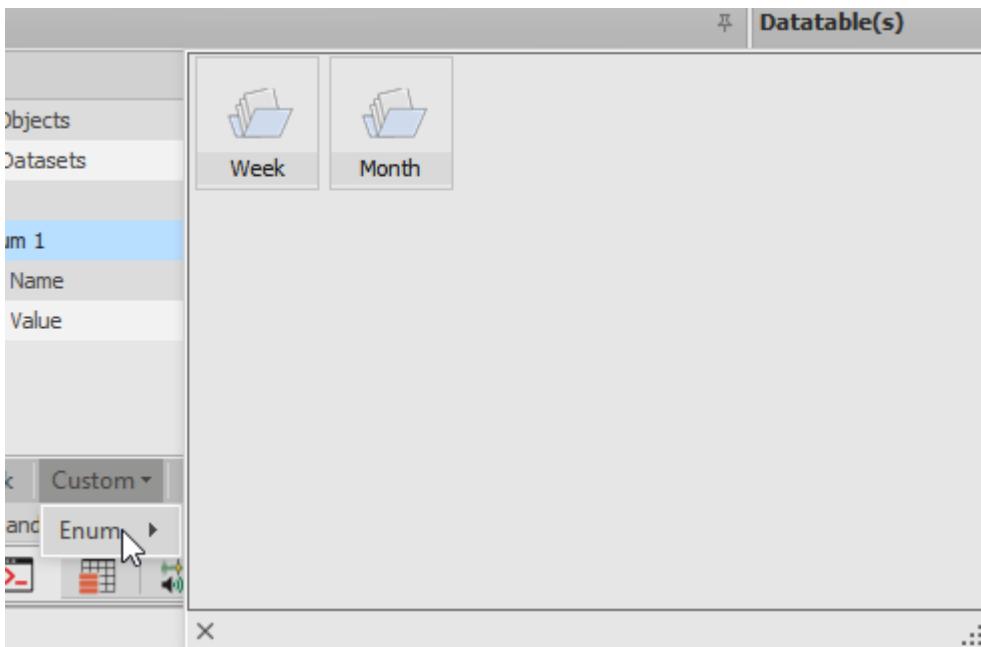
Clock Addin is a versatile tool for creating dynamic and customizable clocks within the Wasp3D render engine. Users can not only design clocks but also control their appearance, time format, and real-time updating. It's a useful feature for broadcasting, visual displays, or any application where accurate timekeeping and clock customization are important.

Custom

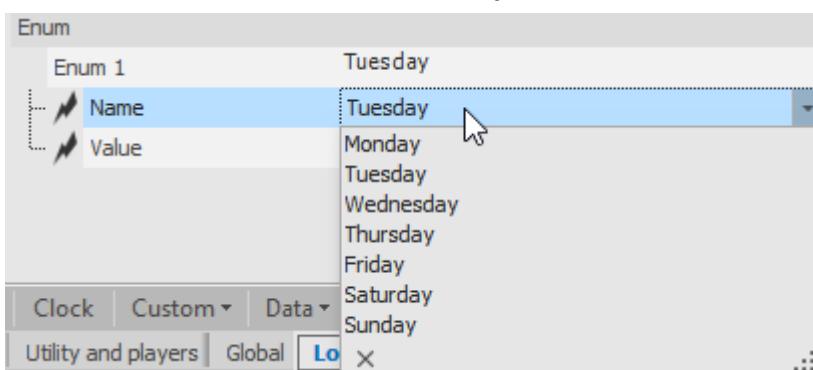
Enums or Enumerations are a common programming construct used to define a set of named constants, making it easier to work with certain types of data that have predefined values.

1. User-Created Enum Lists: Users can create custom enum lists.
2. Mapping Names to Numbers: When a user assigns a name (e.g., "Monday") to an enum value, the add-in associates that name with a corresponding number (e.g., 1). This mapping allows users to work with both names and numbers interchangeably.
3. Mapping Numbers to Names: Conversely, if a user assigns a number (e.g., 2), the add-in will provide the corresponding weekday name (e.g., "Tuesday"). This bidirectional mapping ensures flexibility in how users work with the enum values.
4. Use in Graphics Templates: These enum values (either names or numbers) can be used within graphics templates created in Wasp3D. For example, you could use these enum values to dynamically display the day of the week in a graphic.

By default Wasp3D has provided Weeks and Month enum table which can be used based on Number.

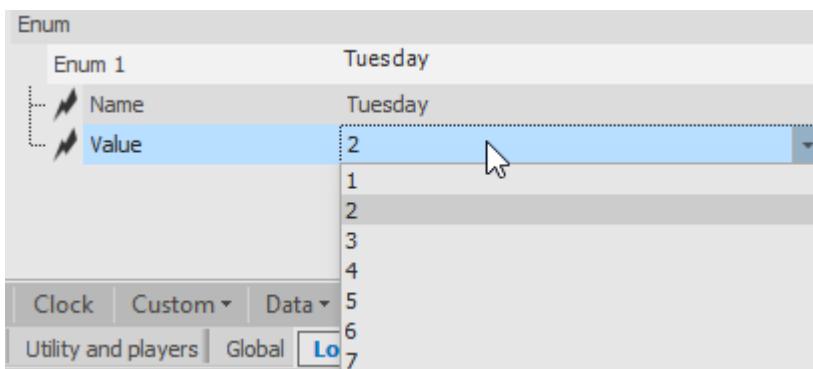


1. Week - Get the names of the weekdays or its number in this enum.

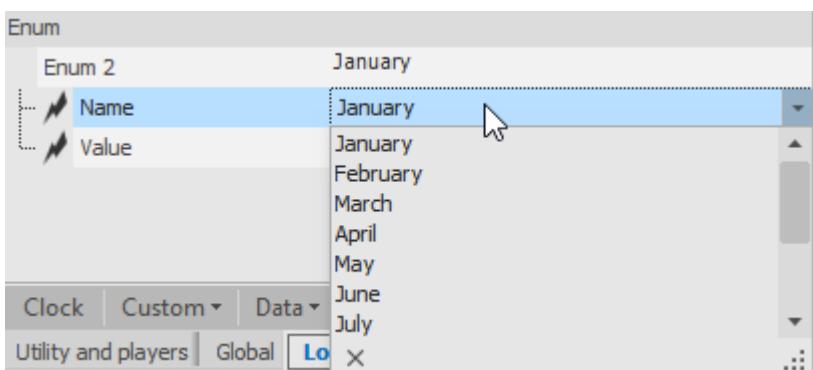


When the user changes a name to the selected Weekday, the Value is set to the corresponding number. for e.g. Monday is 1, Tuesday is 2 and so on.

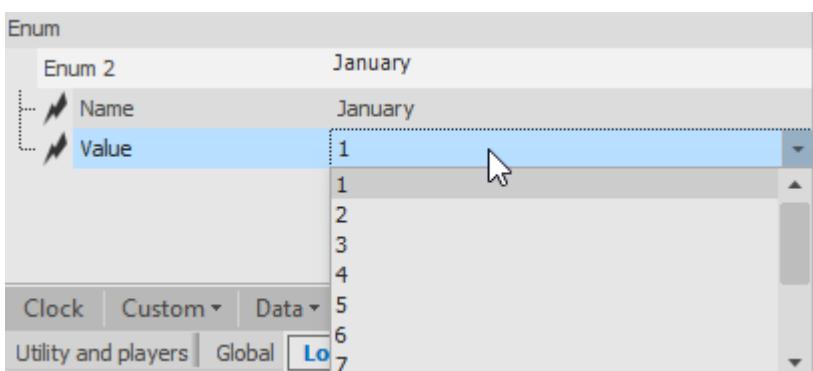
Similarly, if the user changes the value to 1 or 2 the corresponding weekday is provided in the Name string.



2. Month - Get the names of the month or its number in this enum.



When the user changes a name to the selected Month, the Value is set to the corresponding number. for e.g. January is 1, February is 2 and so on.



MultiPin

Allows user to combine data of multiple tables into one as well as add expression based columns to the existing tables. These expressions can be math calculations or filter commands to get data from the selected table. It also allows user to get table data of objects like AI, Tile, Graphs etc. This data is represented in form of values from these objects and provides Min, Max, center position of the objects.

Object Setting - Allows user to define the Tables and Objects from which the Multi-pin ad-din will access the relevant data. It has three columns.

- Column Name - Sets the Name of the column by which it will be added in the multi-pin table.

- Object Name - Allows user to select the object, expression, Variable etc. to access the data
- Object Column - Provides options based on the property selected in the Object Name Column.
- Types of Object Name and their relevant Object Column Settings
 - Expression: Expression allows user to set custom expressions to filter, calculate the data from the selected set of table columns. When expression column is selected in Object Name, the Object Settings column displays the expression window to apply the relevant expressions for the column.
 - Table: Allows user to select data from the same or another table. When selected, it displays Table Selection option in the Object Column. It also allows user to get only the data matching the selected column from the table. For.e.g. there are 2 separate tables one for Name and other master table which has Name, Age, Location, profile columns. So user can get only the Profile Column from the Master table which are matching the names in the selected column.
 - Slave: If a table is selected in any other column, the slave column gets other tables from the data table which are representing the filter applied in table column.
 - Variable: Allows user to select a variable and displays its value in the table. The object column allows user to select the variable from the list of scene or global variables.
 - Counter: Adds a counter column in the table. In the Object Column, it allows user to define the Initial value to start the counter and the increment step value. User can create custom counter table using this property as well.
 - Page Counter: Adds a simple Rank based counter in the table.

Common Settings:

- Data Refresh: no. of seconds in which the data is refreshed.
- Row per page: select number of rows of data to be extracted for single display of data.
- Loop: Repeats the data once reached end of the Table.
- Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
- Update Buffer: When enabled, the data updates are fetched in page and updated on-air. When unchecked, the update is fetched but the text is not updated on-air. The data is updated on-air, when the update buffer action is called using an action set or at required frame on the timeline. The case is useful when the data is wired, but only requires to be updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.

Data: Displays tabs based on sheets available in the UDT. Other sheets are only loaded if there is some data in it. Data is displayed in form of columns. It displays all the columns of the table. User can change the column type using the Arrow drop down displaying different type of data types. The data is displayed as per filter and settings applied in tech common settings.

Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.

- Page: For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.

- Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the change is like this because the 1st item will exit and will become the 6th item, similarly, the send item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.
- Slide: This mode is designed as another menu mode to accommodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required, but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.

Filter: Filters the data table as per the defined filter. Click on the + button to add filter.

To filter table data with static or variable value adding conditions like "begins with", "ends with", "equals to"

- Order by: re-arranges the content of the sheets in the order specified.

Apply Filter: Applies the selected settings to the Data Table(s).

Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify different filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

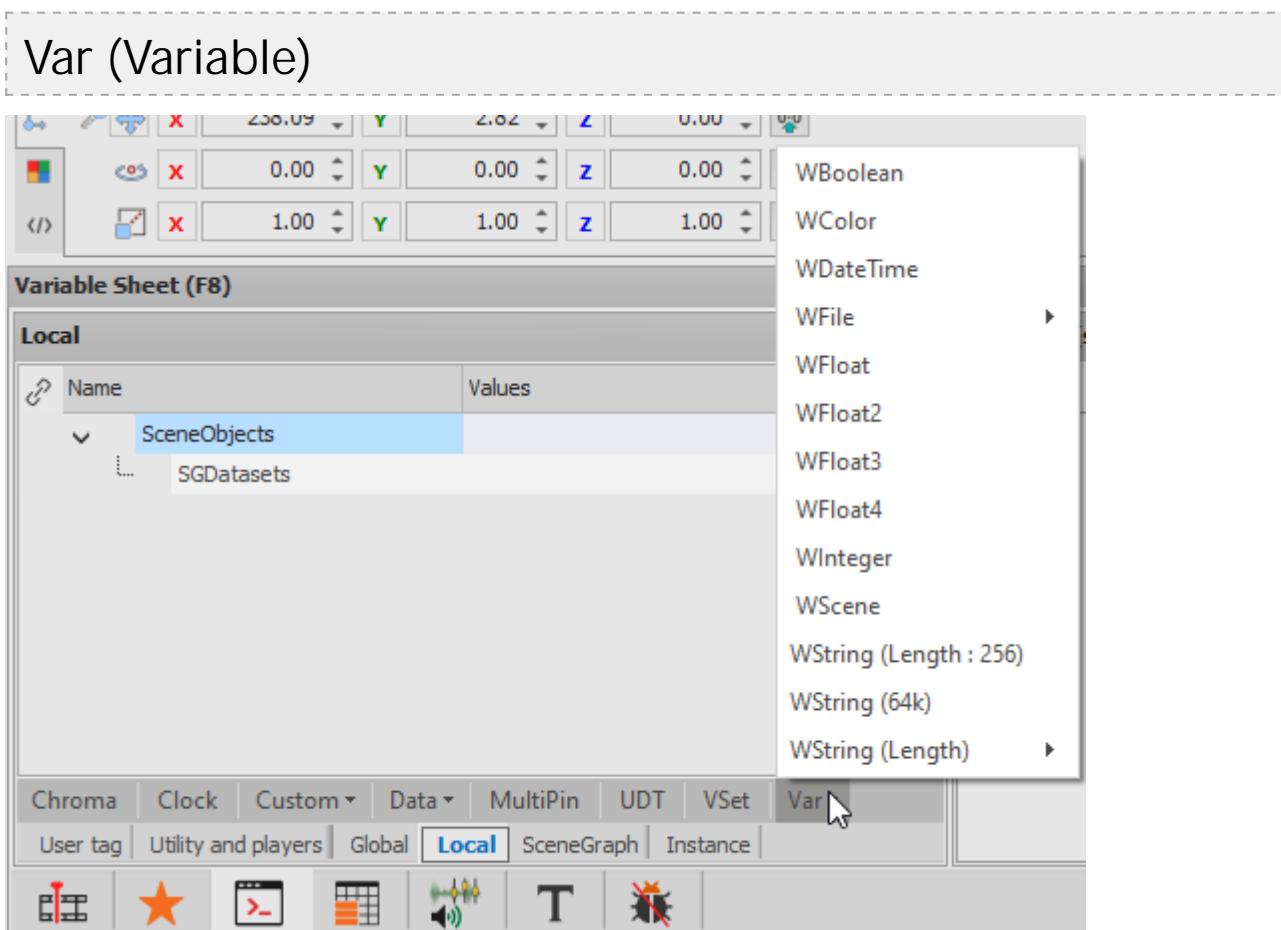
UDT

An acronym for User Defined Tables, designed primarily to facilitate functionality like that of a database. It provides users access to a collection of tables that can be customized to create hierarchical, category-based data entry and storage interfaces.

- Browse: search for the file and add from any location on local system or network systems.
- Common Settings: These settings are applied on all the table sheets in the selected file. To override the common settings user can specify settings per table using the table settings.
 - Row per page: Select number of rows of data to be extracted for single display of data. The Row per page is set according to the number of items to be displayed in one data set of graphic. When user wants to display a team sheet of soccer for 11 players, the row per page would be set to 11. and when showing data for a Map where all constituencies are required to be colored, the All option is selected for it.

- Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.
 - Page: For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.
 - Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the change is like this because the 1st item will exit and will become the 6th item, similarly, the 2nd item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.
 - Slide: This mode is designed as another menu mode to accomodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required, but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.
- Loop: Repeats the data once reached end of the records. The end of records is calculated based on the selected mode.
- Auto Filter: When enabled, filters the UDT as per the Visible Rows.
- Auto Update: When enabled, it updates the data of the currently displayed page on-air, else the data updates are displayed when the data page is called next time or in loop. The updates if done in the rows in 2nd page are displayed when the record move action is called.
- Show Rank: Adds a Rank column in the existing data set. This is useful to see the row numbers and to filter the data based on row number if there no similar column in the data.
- Maintain Schema: Locks the data assignment with the column. This is helpful incase rows are added inbetween the existing data set after the data is mapped to the graphic.
- Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
- Update Buffer: When enabled, the data updates are fetched in page and updated on-air. When unchecked, the update is fetched but the text is not updated on-air. The data is updated on-air, when the update buffer action is called using an action set or at required frame on the timeline. The case is useful when the data is wired, but only requires to be updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.
- Sync: When enabled, the data updated in the table and updated as soon as the user enters the data in the UDT.
- Data: Displays tabs based on sheets available in the UDT. Other sheets are only loaded if there is some data in it.
- Data is displayed in form of columns. It displays all the columns of the table. User can change the column type using the Arrow drop down displaying different type of data types. The data is displayed as per filter and settings applied in the common settings.

- Table Setting: The parameters in Table setting are similar to what is available in the common settings. They are used to override the common settings for the respective table data. Please see the parameters defined in the Common parameters for more info.
- Variables: Displays list of variables in the scene which can be dragged to the filter column to filter the data as per the data point in variable.
 - Local: gives the list of local variables which are only available in the scene in which it is created
 - Global: gives the list of global variables which are available/accessible in any scene
- Filter: Filters the data table as per the defined filter. Click on the + button to add filter.
 - To filter table data with static or variable value adding conditions like "begins with", "ends with", "equals to"
 - Order by: re-arranges the content of the sheets in the order specified.
- Apply Filter: Applies the selected settings to the Data Table(s).
- Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify different filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

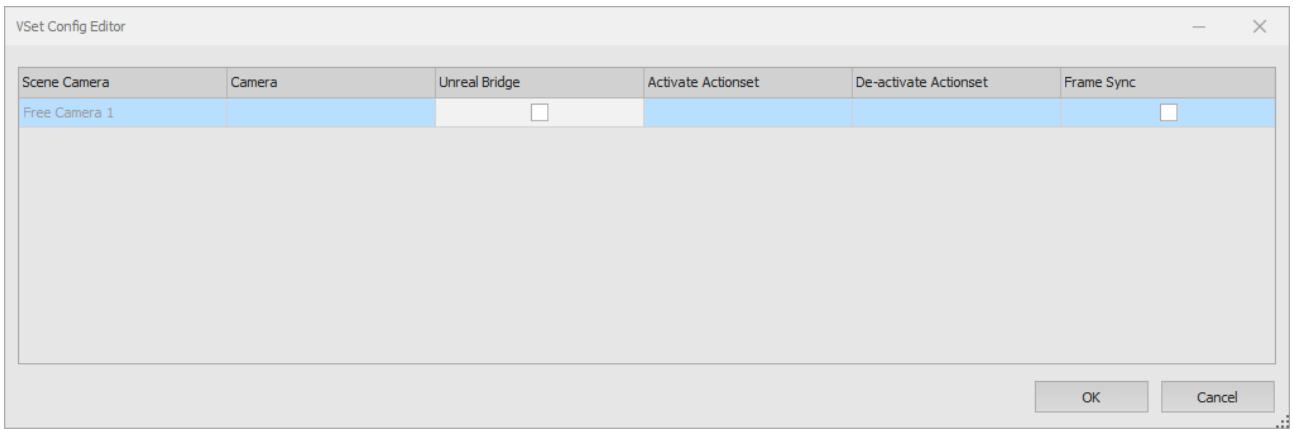


- WBoolean: A data type, having two values (denoted as true and false).
- WColor: Describes a Four-component vector including 4 floats representing Red, Green, Blue and Alpha.
- WDateTime: A variable to set predefined Date & Time.

- Wfile: Creates a File Type variable. The file type can be selected in the dialog box. Following type of Files are supported.
 - Any - all or any type of file to be selected
 - AI Shape - ai or svg files
 - FBX Nurbs - .fbx, .3dm and dae(Collada) files.
 - Import skeletal Mesh - .fbx files with Skeletal Data
 - PSD file - Adobe® Photoshop .psd files.
 - Image - All image files (.bmp, dds, hdr, .jpg, jpeg, gif, png, tga, webp, jfif)
 - Environment - Cubical .dds files.
 - Video - All supported video and audio files (.avi, .mpeg, .mpg, .mov, .mp3, .wmv, .wav, .mx, .mp4)
 - XML - .XML files
 - Excel - All sheet table files (.xls, .xlsx, .ods, .xlsm)
- WFloat: is a decimal based variable which can be wired with float based properties like Radius of a Sphere, Width/Height/Length of a Cube or Decimal Value in a text.
- WFloat 2: Describes a two-component vector including 2 float structures (FLOAT X, and FLOAT Y).
- WFloat 3: Describes a two-component vector including 3 float structures (FLOAT X, FLOAT Y and FLOAT Z).
- WFloat 4: Describes a two-component vector including 4 float structures (FLOAT X, FLOAT Y, FLOAT Z and FLOAT W).
- WInteger: Variables that have whole numbers as their value. These are variables which occupy 4 bytes in memory and can take both positive and negative values.
- WScene: This variable gets the scene id or instance id from the Playlist Addin to perform Waspi Actions.
- WString (Length): String is a data type that represents sequences of characters or text-data. String 8K variable supports character length of up to 8192 characters in a single string.
- WString (64K): String is a data type that represents sequences of characters or text-data. String 8K variable supports character length of up to 65536 characters in a single string.
- WString (Length): String is a data type that represents sequences of characters or text-data. User can specify a limit to the number of characters of the string using the length WString variable.

VSet

V Set addin allows the Wasp3D Scene to connect with V-Set Application. All scenes with v-set addin are supported to run as Virtual Set or Augmented scenes.



User can connect scene camera with cameras specified in Virtual Set application. When the scene is connected with Camera 1, 2 and 3, they can be set as Tracked Cameras in the V-Set Addin.

- Free Camera - Displays the list of Cameras in the scene. User can select a scene camera to map with the Camera specified in the Camera Configuration of Virtual Set application.
- Camera - Displays the list of 8 Cameras supported by Wasp3D Virtual Set Client application. The configuration of each camera can be controlled in the Camera Configuration of Virtual Set application.
- **Unreal Bridge - When enabled**, links with the respective Wasp3D Scene camera with Unreal Engine Camera. Unreal Engine needs to be configured before selecting this option.
- Activate Actionset - An Action Set can be selected to trigger whenever this camera gets activated.
- **De-Activate Actionset** - An Action Set can be selected to trigger whenever this camera gets de-activated.
- FrameSync - By Default all Action Sets are run Async, When Frame Sync is enabled, the Selected Action Sets in Activate or De-activate are run in sync with the render frame.

Global Variables

Global Variables: are available across multiple scenes. User can create a global variable and bind multiple scenes with the same variable. For e.g. Change the color of an object in all templates using just one Color Variable in Global Variables Pool. The Global variables are supported per Engine and not across the network.

User can create a variable in Local variable pool and then Drag-Drop it to the Global Variable Pool to make it global.

Following variables are predefined global variables:

- Back Buffer Height: displays the height of the engine's render area.
- Back Buffer Width: displays the width of the engine's render area.
- Frame Rate: displays the number of frames being rendered per second in the engine.
- Interlace: displays if the rendered output is interlaced. Checked-ON indicates interlace true.
- Engine Name: Provides the name of the Sting Server to the scene. This is helpful if the user wants to change the text of the scene based on the Engine it is played. For e.g. User creates one

template with option to show text either in Hindi or in English, so he can create a Match Variable and assign actions based on the engine name.

Instance

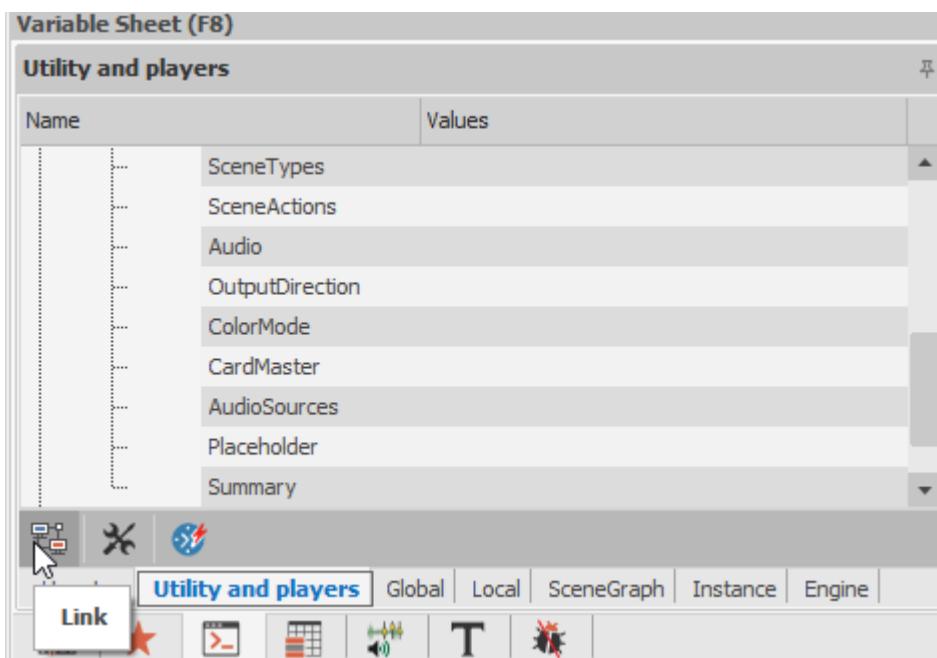
Scene Graph

User Tag

User Tag is a descriptive label, or a name tag used to define a field as variable and to provide the flexibility to change its parametric values from an external source (e.g. database, wire feeds etc.). It can be assigned to both interpolated (key frames) and non-interpolated (text, textures etc.) parameters and objects.

Utility and Players

Utility and Players is a Drone Designer add-in, which allows extending the application features using Skeletal Addin, Communication Addin etc.



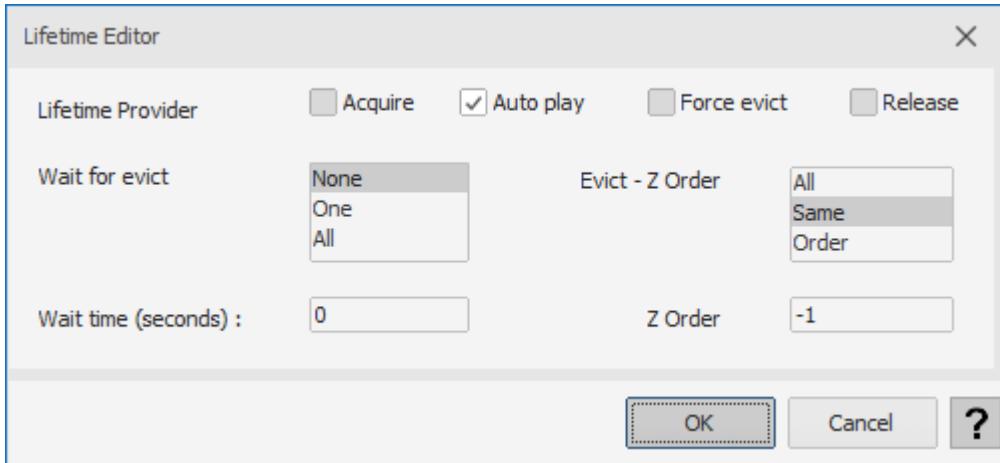
Link :



Tool Addin :



Lifetime Editor:



Lifetime Provider

Active: When checked ON, activates the Lifetime Addin in the template.

Acquire:

AutoPlay: When checked ON, automatically releases the template from 'Pause' state when "Cleared to Play" trigger is received.

Force Evict: When checked ON, this template will render off & unloads all the templates & will send a "Cleared to Play" trigger.

Release:

Wait for Evict: Will let the user decide the time frame to perform the action.

- All: Will wait for all the templates to unload/evict.
- One: Will perform the action, on evicting any or one of the templates.
- None: Will not wait for "cleared to play" trigger. It will perform the action simultaneously.

Evict Z-Order

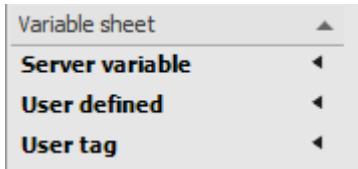
- All: All the templates loaded on sting server will be affected.
- Same: It will affect the template with the same Z-Order of that of the raiser template
- Custom: Can define a specific number, or can filter on the following basis, <5, >5, 5, 0-15 (Greater than, Less than, Equal To (5), In between, comma separated values).

Wait Time Seconds: It will perform the action, after the given time frame (in seconds). Even if it doesn't receive any trigger.

ZOrder:

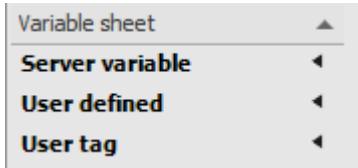
Server Variable

Created as well as located on the Kernel Controller (KC), the Server variable is accessible to all connected to KC. The Server variables can be used in multiple scenes to show the same data simultaneously.



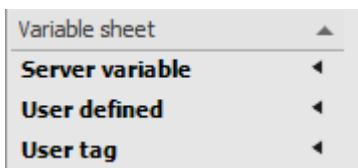
User Defined

Created as well as located only in the current scene, User variables work only for the scene in which they are created. The User variables need to be wired with the parameters to make them dynamic.



User Tag

Created and tagged with Parameters (Interpolated and Non-Interpolated) that need to be made dynamic while the scene is On-Air. User tags can be best used as array to tag multiple parameters at one time.



Hierarchy Pool

The Hierarchy window displays all objects in a hierarchical order as defined in scene. Objects can be in a Group based hierarchy or a Parent-Child based hierarchy. It has 3 Tabs, Hierarchy, Views and Metadata.

Do any of the following to open this window:

- On the Window menu, **click Hierarchy** Or Press (F9) from the keyboard.

Hierarchy (F9) ?

Hierarchy		Views	Quick Acess 1	Quick Acess 2	Quick Acess 3	Render	Tag
	Node 7	Material 8				10	11
1	2	3	4	5	6	Scenegraph	
<input checked="" type="checkbox"/>		Background	None	M			Back buffer
<input checked="" type="checkbox"/>		Layout	None	M			Back buffer
<input checked="" type="checkbox"/>		Directional Light 0	None		255,255,255,255		Back buffer
<input checked="" type="checkbox"/>		Free Camera 1	None	45.00	Perspective	5,000.00	Back buffer
<input checked="" type="checkbox"/>		S_Sphere 1	M	20	0	100	Back buffer
<input checked="" type="checkbox"/>		S_Cube 1	M	Centre	0.00	1	Back buffer
<input checked="" type="checkbox"/>		U_Text3D 1	M	3D Text	Free	C nc Default	Back buffer
<input checked="" type="checkbox"/>		S_N_Gon 1	M	3	1.00	1.00	Back buffer
<input checked="" type="checkbox"/>		Circle 1	None	M	20	0.00	Back buffer
<input checked="" type="checkbox"/>		Star 1	None	M	5	5	Back buffer
<input checked="" type="checkbox"/>		Callout 1	None	M	Bottom Right	40.00	Back buffer
<input checked="" type="checkbox"/>		Ellipse 1	None	M	20	0.00	Back buffer
<input checked="" type="checkbox"/>		Gear 1	None	M	6	0.00	Back buffer
<input checked="" type="checkbox"/>		Star 2	None	M	5	5	Back buffer

Default

Default None 0,0,0,0,100,0,100,00 Default

There are 11 columns in Hierarchy:

1. Check Box - Shows whether the respective object is part of the respective view or no.(To know more about Views and Canvas, please visit the Views Tab help)
2. Object Class Icon - This icon is shown based on the class of the respective object.
3. Trigger Icon - Shows a Lightning icon representing Gesture. A gesture icon will be displayed along the objects with gesture(s). User can drag and drop this icon to another object to copy the gestures from one object to other. Click on the Trigger Icon to assign events to the respective object.
4. **Animation Key Icon** - This column is shown blank until an animation is applied to the object. Once the animation is applied, Shows a Key icon representing animated objects. User can drag and drop this icon to another object to copy the animation from one object to other.
5. Translate Icon - Represents object translation. Users can drag and drop this icon from one object to another to duplicate the object's translation onto the new object.
6. Hide and Unhide object - This is a button to Hide/Unhide an object. User can press the eye icon to Hide and Unhide the respective/Selected object. The icon changes to open or a closed eye based on the relevant operation. The Hide and Unhide works hierarchically. If a parent is hidden, the child objects also hide. But user can select the child object to Unhide as per requirement by individually selecting the required operation.
7. Node - Shows the Names of the objects. User can rename the object by pressing F2 when the respective object is selected.
8. Material – Shows the material assigned to the object. User can change the material of the object by clicking on the material to display a combo-box list of all the materials made in the material pool.
9. **Quick Access (1,2 & 3)** - The Quick Access parameter for specific object classes is displayed here. All objects in Wasp have their frequently used properties set as Quick Access parameters,

allowing users to conveniently modify these properties directly from the Hierarchy window. Users can mark up to 3 favorite parameters as Quick Access (Quick Access 1, 2, or 3) in the Hierarchy Dock by right-clicking on the respective object's parameter in the Parameter Pool and selecting the desired Quick Access column.

10. Render - Shows the Render Target on which the respective object is being rendered.
11. Tag - Shows the tag assigned to the object. Tags can be used to store a value for the respective object, It can be name of another object or any value. User can use the Self.Tag object for the object to quickly use this value from the respective object.

User can change the order of display of column by simply click and drag the column from the label bar and move to required column location. Apart from the 1st 6 columns all other columns can be ordered as per requirement.

Select Multiple objects and press "CTRL+G" to group the selected objects into one group.

Drag and Drop the child object to parent object to form a parent-child hierarchy.

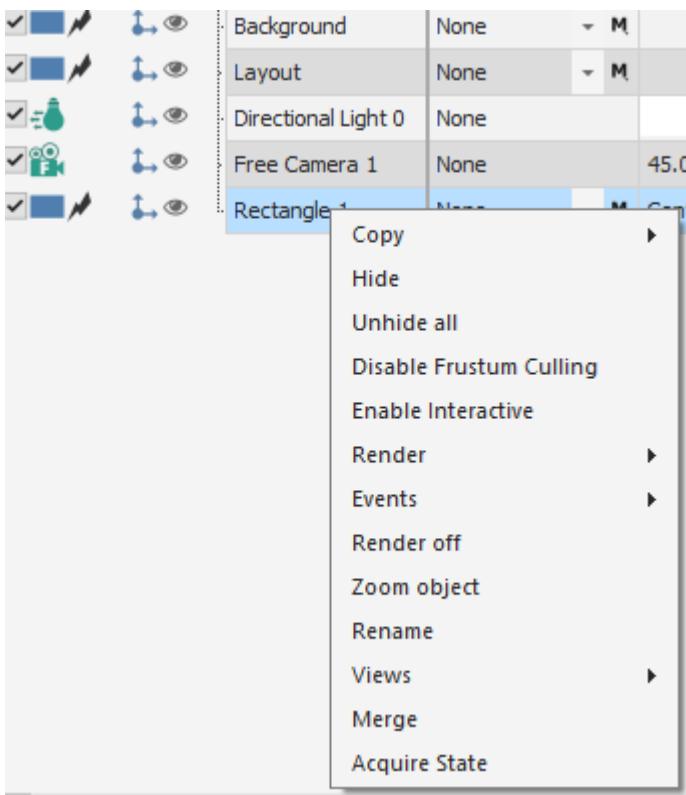
Expand a collapse hierarchy to view the objects in the selected group. "CTRL+Expand" would expand all hierarchy chains under the selected object. Similarly, "CTRL+Collapse" would collapse all the hierarchy under the selected object.

Right Click on the Scene Graph (on top of the Hierarchy pool) to see more option which can be assigned to the scene graph or objects in the scene.

- Unhide All – Unhide all the objects in the scene.
- Events – Provides access to assign an event to the scene.
 - On Trigger – occurs when the selected trigger is raised in the scene or from another scene. User can execute actions when a specified trigger is received by scene.
 - On Named Event – occurs when a named event is received by scene. User can specify the Named Events in the Scene Graph [Named Event] window based on which the specific actions or action sets are to be executed.
 - On Before Prepare – occurs just before the scene starts to prepare to render in Sting Server or Waspi Mimosa. This event is helpful to assign or set values to the dynamic elements before start to render the scene from '0' frame.
 - On Evict - Function executes a defined list of actions or action sets when a scene is unloaded. For example, it can automatically trigger another graphic action when the current scene is unloaded. It is also triggered when received from another template with lifetime add-in, to execute the set of actions before the scene unloads.
 - On Scene Unload – Occurs when the respective scene unloads or simply saying after Play Complete is executed. User can raise triggers or trigger next graphic or execute actions as per requirement.
 - On Scene Cue – Occurs when a scene is cued and starts to render on the Sting Server or Mimosa. The scene stays on frame 0 until it is played.

- Cleared to Play – Occurs when another scene using Life-Time AddIn is unloaded from the Sting Server and send the command to play "next graphic" or "play complete". This event is primarily used with Life-Time AddIn.
- On Before RT - Event is executed just before a field is rendered for a Render Texture created in the Scene.
- **On Prepare** - Performs actions when the instance is loaded in the Playout Engine and ready to Cue in the Playout Client. For example, if you have a countdown timer in your scene, it will start running as soon as the instance loads, ensuring that everything is set up and ready before the scene is fully displayed.

Right-Click on an object to get Additional functions for the respective object.



- Copy – There are 4 properties which can be copied from an object to another object. User can copy these properties from one object and paste these to other objects. As soon as you copy Animation, Transformation and Triggers (gestures/events) from an object, Paste Animation, Paste Transformation, Paste Triggers options get available in the Right-Click context menu.
 - Object - Copies the object itself. Makes one Copy at a time. Press Shift with Copy to get dialog box to Copy multiple objects and assign transformation as well as Copy Materials, Copy/Clone Textures, Copy Events of the respective object to its copies.
 - Animation - Copies the Animation tracks of the object. Once the object's animation is copied, Paste Animation option is available in the context menu to paste the copied animation. Remove Animation option also get available if the object is having an animation applied to it.
 - Transformation - Copies the Transformation (Scale, Rotate, Translate and Visibility) of the object. Once the object's transformation is copied, Paste Transformation option is available in the context menu to paste the copied transformation.

- Events - Copies the events/triggers and actions applied on the respective object. This option only gets highlighted when gestures are applied to an object. Paste Events option is available in the context menu to paste the copied events.
- To clipboard - copies the object to the scene's clipboard. User can use this command to copy an object from one scene(s) to other without using the Favorites Pool.
- Hide – This is a toggle menu option to 'Hide' or 'Unhide' an object. Press Hide object to hide the object from the viewport. Unhide option to Unhide the object from the viewport. The object is still being rendered when it is hidden.
- Unhide All - Unhide all the objects in the scene.
- Disable Frustum Culling - Frustum culling enhances rendering performance by excluding objects beyond the camera's visible frustum area. This space resembles a pyramid and determines what's visible. By removing objects outside this range, it reduces needless rendering, enhancing efficiency.

To manage frustum culling, users have the option to activate or deactivate it for the entire scene using the Frustum Culling checkbox in the Scene Graph variables. Alternatively, they can opt to deactivate it selectively for specific objects by employing the "Disable Frustum Culling" property, impacting solely the chosen object without affecting the entire scene. When the "Disable Frustum Culling" property is applied to an object, the object's name is shown with slanted lines. This property can also be accessed from the Parameters Window within the Display Properties Tab.

- Enable Interactive - Once "enable Interactive" is enabled to an object, it becomes responsive to gestures and events, making it accessible for coding purposes. Activating this option readies the object for recognition within both the Code and Scene environments, where it will function as an interactive object. Additionally, upon activation, a Hand Finger Gesture icon is added before the object name.
- Select Material – This option will select the material assigned to the respective object in the Material Pool. This is helpful when the user does not want to search the material of the respective object and saves time.
- Render - All objects are rendered to the "Back Buffer". However, when Render Target textures have been created, then objects in scene can be rendered to the Render Texture(s). This is used to create effects like Mirror reflections, God Rays etc.
 - Back Buffer - When checked the selected object(s) will render to Back Buffer. This option is checked by default.
 - Render Texture - User can check-on this option to render the selected object(s) in the respective Render Texture.
- Events - Events or triggers can be assigned to objects.
 - On Gesture – occurs when the user creates a Touch gesture in Waspi Mimosa interface. There are 13 Simple Gestures and 26 Complex gestures to which a user can assign action(s) or Action Set(s).
 - On Mouse Down – occurs when a mouse is tapped for transformation in the scene.
 - On Updating – occurs concurrently until the time the respective object is transformed in the scene.
 - On Updated - This event is triggered when an object has finished updating or has been moved to a new position. It occurs after the object's transformation is fully completed, signaling the end of any updates or changes in its position or properties.

- On Scroller – occurs on multiple levels depending upon the position of each Scroller Group in the ‘Scroller Node’ ‘Displacement Area’ region. There are nine type of Scroller events **Head In, Head Out, Tail In, Tail Out, Head Marker 1, Head Marker 2, Tail Marker 1** Tail Marker 2 and Pre Roll. User can control the movement of the scroller as well as execute actions based on these scroller positions. The occurrence of these events depend upon the direction of the scroller. If the scroller is moving from left to right, the head of the scroller will be the right edge and Tail will be the Left Edge of each ‘Scroller Group’ and vice-versa. These events will be triggered multiple times depending upon times each ‘Scroller Group’ have entered the ‘Scroller Node’ region.
 1. Head in – occurs when Head of the ‘Scroller Group’ enters the ‘Scroller Node’ region.
 2. Head Out – occurs when Head of the ‘Scroller Group’ exits the ‘Scroller Node’ region.
 3. Tail in – occurs when the Tail of the ‘Scroller Group’ enters the ‘Scroller Node’ region.
 4. Tail Out – occurs when Tail of the ‘Scroller Group’ exits the ‘Scroller Node’ region.
 5. Head Marker 1 – occurs when the Head of the ‘Scroller Group’ reaches the position of Marker
 6. Head marker 2 – occurs when the Head of the ‘Scroller Group’ reaches the position of Marker 2.
 7. Tail Marker 1 – occurs when the Tail of the ‘Scroller Group’ reaches the position of Marker 1.
 8. Tail Marker 2 – occurs when the Tail of the ‘Scroller Group’ reaches the position of Marker 2.
 9. **Pre Roll** – This event is fired just before the entry of the object according to the Pre Roll distance defined in the scroller node parameters.
- On Scroller Complete – occurs on Head-In of the last record of the scroller.
- On Timer Complete – occurs when the Timer object reaches an interval. Periodic events are received multiple times and one shot event is received one time. It also occurs when a Text3D object in Countdown or Stopwatch mode completes its time.
- On Page Change – occurs when there is a change of page in a ‘Page Controller’.
- On Menu Change – occurs on multiple levels depending upon the position of each menu-item in a “Menu Group”. There are three type of Menu Events – Node Enter, Node Exit and Content End.
 1. Node Enter – occurs when a menu item is rendered on in the scene at the start of its trajectory (not Visible On).
 2. Node Exit – occurs when a menu item has completed its animation trajectory and is rendered off from the scene.
 3. Content End – occurs when the data source of the Menu Group data table is at last page and the last menu item has reached the menu extent.
- On Variable Change – occurs when there is a change in the value of a variable
- On Jump – occurs when the Jump action is completed in a Menu group of any object which is scrubbable.
- On Data-Row Change – occurs when there is change in data of the current page of the tabular Data Addins. This is valid for all Data Set type Addins like UDT, Excel, Finwiz, ODBC etc.

- On Scene Prepare – occurs when the respective scene is prepared to cue in the render engine.
- On Scene Unload – occurs when the respective scene unloads or simply saying after Play Complete is executed.
- **Render Off** – is a toggle menu item. It renders off the selected object(s). When an object is rendered off, it saves the render time of the scene.
- **Zoom Object** – Gets the selected object in focus of the viewport. It works only in orthogonal views like Top, Front, Left Right etc.
- **Rename Object** – Provides a simple tool to rename the selected batch of objects. User can select to rename Material and Textures associated with the object by clicking on the checkbox and providing the name of the respective property.
- **Views** – Shows the list of Views created in the scene and allows user to assign the respective object(s) to the view(s). User can add or delete the object from the view using this option. If user selects an object and assigns it to a view, it will only be displayed in the selected view. If an object is not part of the active view(active view shown as clicked in the View Bar).
- **Acquire State:** Below the hierarchy list, is the Views bar. It shows the views which are created in the scene and shows a button to go to a respective view and see the object assigned to it. User can click on this button to show the object hierarchy as per the selected view.

Below the hierarchy list, is the Views bar. It shows the views which are created in the scene and shows a button to go to a respective view and see the object assigned to it. User can click on this button to show the object hierarchy as per the selected view.

There are 2 buttons on the right side of the View bar to show or hide the objects for the respective view. Layout, Background, Directional Light, and Camera are always active for all views.

1. **Show all objects** – Displays all the objects as per the default hierarchy. But the objects which are not part of the view are displayed as disabled.
2. **Show View objects** – Shows objects which are included in the respective view.

Views Tab has the following options

Hierarchy		Views			Meta data
					New
Render Seq	Name	Camera	Position	Canvas	
	Default	None	0.00,0.00,1...	Default	

User can assign size and position to a view and render the objects of the respective view to the desired position and size. This is helpful when user wants to render multi-camera renders to one viewport or show Picture in Picture kind of scenarios in the same scene. The View on the top of this list gets rendered first. If objects of one view needs to be displayed in another view. The respective view needs to be rendered first so that its objects can be displayed in the next view. Render the View 1st to display in the next View. This order is not directly associated with the Canvas on which it is being rendered.

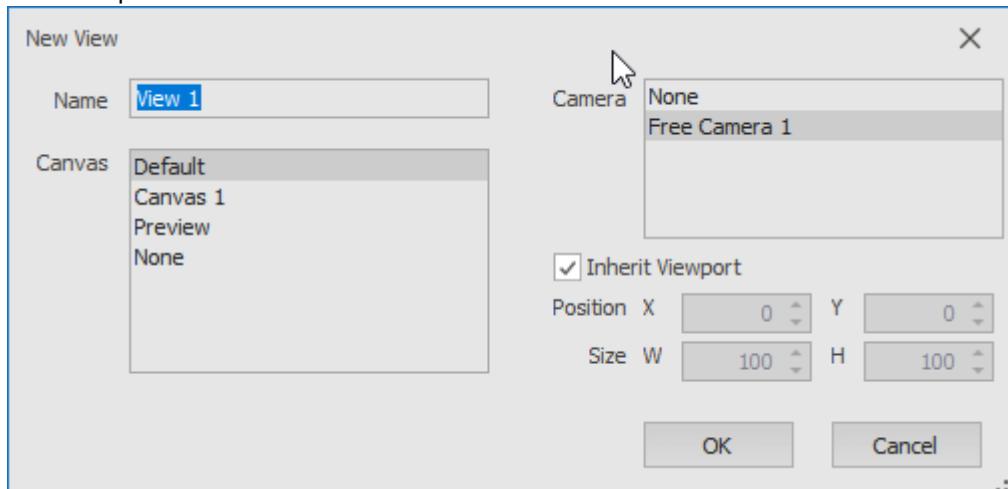
Views Tab has 6 columns

1. **Eye** – Hides and reveals the view in the selected canvas.
2. **Ordering Arrows** – Allows user to define the render order of the added view.

3. **Name** – Enter the name of the new View here. User can press "F2" to change the View name in the View Grid also.
4. **Camera** – define a specific camera to render the view. If None is selected, the active camera for the view is selected to be displayed.
5. **Position** – is a Vector 4 control. Sets the Position and Size of the rendered view in respect to a 100 x100 unit based grid. To do this precisely, user can divide the 100 units by the width or height of the canvas to get the X and Y values respectively. Set the size of the render area for the respective view. User can set W – 100 and H – 100 to full size of the canvas. and W – 50 and H – 50 to render half size of the view.
6. **Canvas** – define the canvas to which the would would get associated. This allows the objects of the respective view to render to the selected canvas.

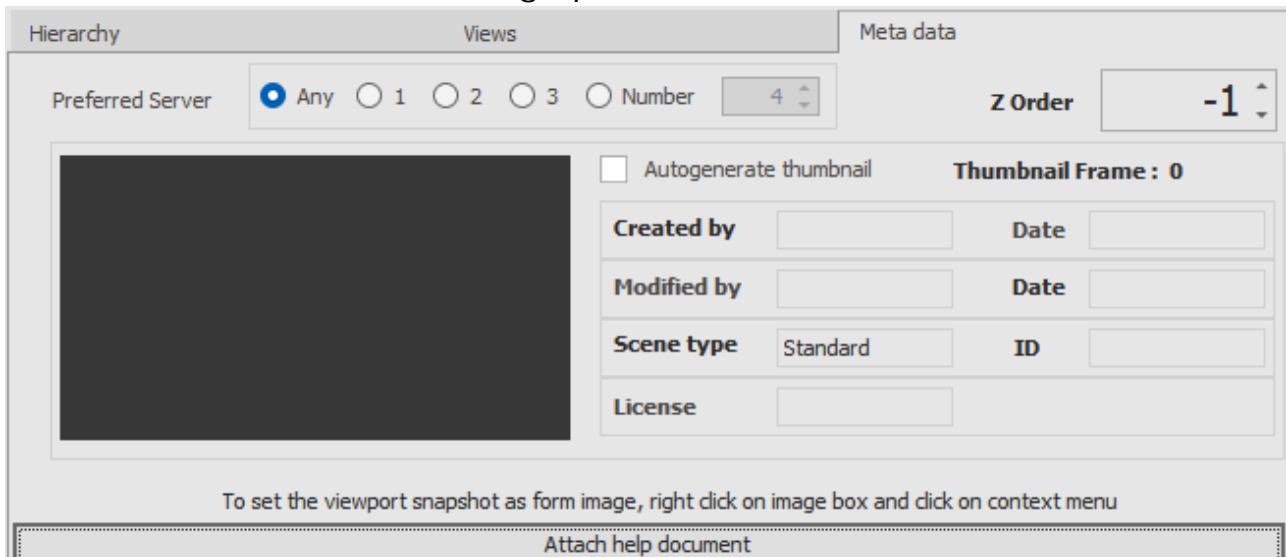
By default Default view is created in the scene and cannot be deleted. User can create more views as per requirement.

New – Opens the New View window



- **Name** – Enter the name of the new View here. User can press F2 to change the View name in the View Grid also.
- **Canvas** – define the canvas to which the would would get associated. This allows the objects of the respective view to render to the selected canvas.
- **Camera** – define a specific camera to render the view. If None is selected, the active camera for the view is selected to be displayed.
- **Inherit Viewport** - When Enabled, the respective is set to the size of the Default Viewport. When Disabled, the Position X Y and Size W H boxes are enabled to define a custom size of the respective view.
- **Position X and Y** – Set the Position of the rendered view in respect to a 100 x 100 unit based grid. To do this precisely, user can divide the 100 units by the width or height of the canvas to get the X and Y values respectively.
- **Size W and H** – Set the size of the render area for the respective view. User can set W – 100 and H – 100 to full size of the canvas. and W – 50 and H – 50 to render half size of the view.

Metadata Tab has the following options:



This option sets the Z-Order of the scene. Users can choose to load the template at a predefined Z-Order, which is particularly useful when multiple templates need to be loaded from the same playlist and displayed in the same screen area.

- **Thumbnail Frame:** Displays the specific frame number where the corresponding thumbnail is captured. This allows users to identify the exact frame that was used to generate the thumbnail.
- **Thumbnail Box:** The Thumbnail Box allows users to manually generate a thumbnail for the template. By double-clicking inside the Thumbnail Box, users can capture a snapshot of the current viewport frame and set it as the thumbnail for the scene. This thumbnail is saved with the scene. By default, the thumbnail is automatically generated when the scene is saved at a specific frame. However, users can override the auto-generated thumbnail by manually creating one through the Thumbnail Box. When a custom thumbnail is created, the "Auto Generate Thumbnail" checkbox will be unchecked.
- **Auto-Generate Thumbnail:** This checkbox is enabled by default to automatically generate a thumbnail for the scene at the time of saving. If users manually create a custom thumbnail by double-clicking in the Thumbnail Box, this checkbox will be unchecked, indicating that the automatic thumbnail generation is no longer active. Users can re-enable automatic thumbnail generation by checking this box again, which will capture a new thumbnail the next time the scene is saved.
- **Created by:** Displays the username of the person who initially created the template. This entry is permanent and does not change once set.
- **Created Date:** Shows the date when the template was originally created.
- **Modified by:** Displays the username of the last person who modified the template. This value changes based on who made the most recent changes to the template.
- **Scene Type:** Indicates the type of scene, such as a Virtual Scene or a Standard Scene. Most scenes created by users are typically standard scenes.
- **Modified Date:** Shows the date when the template was last modified.
- **ID:** Displays the scene's ID if it has been assigned by the user.
- **License:** Displays the name of the licensee associated with the scene.
- **Attach Help Document:** The Attach Help Document button opens a dialog box that allows users to upload a PDF file alongside the scene. This dialog box also provides the option to view the

attached PDF. The document is saved with the scene and can only be accessed within the design interface. Users can use this feature to attach "Read Me" files or help documentation for others working with the template, facilitating easy sharing of instructions or important information related to the scene.



Note: Channel name should be specified in Sting Server.

Ad Log

Is used to define and execute data logging tasks. It exports data functions for offline analysis in third party applications like Microsoft Excel. It records and observe real-time data display.

- Advertisement ID - Is used to define a Constant, Variable or Data Source to specify the advertisement ID in the Advertisement Log recorded in the Database. It can be any Number or Text String by which the specific Advertisement can be identified
- Advertisement Slug - Is used to define a Constant, Variable or Data Source to specify the advertisement slug in the Advertisement Log recorded in the Database. It can be any Number or Text String by which the specific Advertisement can be identified
- Duration - Is used to define a Constant, Variable or Data Source to specify the duration of the advertisement in the Advertisement Log recorded in the Database. If not specified, a default value of 20 seconds is added to the advertisement log.
- Preview Mode - When checked, enables Ad Log to be generated in Preview Mode also. Note Preview Program functionality in Sting Server should be implemented.

Play AnimSet:

It starts animation set. It takes the track of the Targeted Object and Animation track as input parameters. This method is also useful in releasing a pause in Animation set which has already been applied to target node's track.

- Target- Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
- Animation Sets- Displays list of Animation Sets in the scene.
- Copy AnimSet- Is used to create a copy of the selected animation set at runtime. The copied Animset is not saved in the scene. It is used to apply same animset at a time to multiple objects with different key values, often used in programming scenarios.

Stop AnimSet

It stops animation set and remove its effect from the target node's track. The param retain transform defines, if on stopping this animation set the target node will retain the last value of the applied animation or revert back to its previous value.

- Self/Target- allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
- Animation Sets- displays list of Animation Sets in the scene.
- Retain Transform- enables or disables, whether the object (on which the animation set is applied) stays at the new animated state or reset to the last state.

Live Controller

Is used to adjust the volume of the live video.

- **Target -**

- Channel Info- Is used to select the Input channel of the Video (Channel 1 as A, 2 as B, 3 as C or 4 as D).
- Volume Level- Is used to define the loudness of the live video streaming at the selected Channel ID.
- Transition Time (in frames)- Is the time duration for the volume to tune to the specified level.

Switch Camera

Sets/switches to the specific camera selected from the list provided, as the active camera in the scene at any point in the design.

- Choose Camera - lists the cameras in the scene, including the orthographic views (Front, Back, Left, Right, Top, Bottom).

Unreal Cam Bridge

Sets/switches to the specific camera selected from the list provided, as the active camera in the scene at any point in the design.

- Choose Camera - lists the cameras in the scene, including the orthographic views (Front, Back, Left, Right, Top, Bottom).

Send

Send option allows users to send a single command through the communication link for communication purposes.

Start

Stop

Check Variable

It acts as gateway to the next (set) of actions in stack. If the condition is true, it allows the execution of these actions else, it skips them.

- Target - allows user to select the objects, variables or data sources based on which the check variable function will be executed.
- Default – Allows user to build up a default case where value of a variable, data source or object PARAM can be checked to different conditions like: If the variable value is “EQUAL TO”, a constant value 10, then perform certain actions, else ignore the action stack and move on.

Match Variable

It acts like a switch case command. The defined expression is compared with the values of each case, If there is a match, the associated set of actions are executed else the Default action is executed.

Target

- Object - Is used to select the object, its parameters or its transformation, variables or data sources to be used as a source in the expression.
- Variable - Is used to select a UserDefined, Server or System variables, which can be used as a source in the expression.
- Data Source - Is used to select a cell of UDT, SQL, XML or Excel addin tables, which can be used as a source in the expression.
- Add Case - Cases are created by clicking on this button and specifying the conditions.
- Exit Switch Case - Once the cases have been created the exit switch case button preserves them within the Match Variable.

Record Move

Is used to go to next, previous or a specified page of the data table, of a Data (UDT, SQL, XML) based Addin, if pages are available it is a success and will execute the actions stacked under SUCCESS, else it will execute the actions stacked under FAILURE.

- Target: Allows user to select the data source
 - 1. Next - Switch to next page of Data Addin.
 - 2. Previous - Switch to previous page of Data Addin.
 - 3. Page - Enables user to switch to a specific page of the Data Addin based on the values selected as a Constant, from a Variable or Data Source.
- Success: A condition where a stack of specified set of actions will be executed in the presence of data
- Failure: A condition where a stack of specified set of actions will be executed in the absence of data/at the end of the data.

Reset UDT

Script

Is used to control animations or apply actions using simple Java Scripts based on the object, parameters, transformations, variables and data sources defined in aliases which are then used as a,b,c, etc. in the script.

- Library - Contains a list of Pre-made scripts which can be used to achieve the desired results
 - For e.g. -**
 - GetDayOfWeek - This script will get the day of the week from the specified date Create two 3D Text object and DateTime variable, Wire the Date Time variable to 3D Text and get the long date value number as Alias A and Text parameter of 2nd 3D Text in Value B.
 - TranslateNode - Run the script and see that, the Value in 3D Text 2 is the weekday as specified in Date Time variable. change the date and run the script again to see the change.
 - Time Ago -
 - **Set AI Poly -**
 - **Math** - Provides a coding interface in which the user can write the java script based on the aliases defined as a,b,c etc.
 - Increment -
 - Decrement -
 - Is Even Number-
 - Aliases - Displays the objects and data addins in the scene which the user can define as variables in the script
 - Object - Lists the objects, Materials, User Defined, System and Global variables in the scene.
 - Addin -
 - Event Programs -

- Script - Provides a coding interface in which the user can write the java script based on the aliases defined as a,b,c etc.

Send Status

This action works in sync with the Status Control or Salvo button control object in the Data tab. It sends the message written in the control key to any or both controls. The messages are viewable on the Player Control in the Sting Client. User can add Status Control control and Salvo Button on the Player tab of Standard form or directly on Unified form.

- Salvo - Check on if user wants to associate the Send Status message to Salvo Button. This also enables addition controls defined below.

Message Tags: A Message tag is like a place holder for carrying the message to the status control. User needs to define the Message Tag in the list to start sending messages to the Status Control in the Player. Press F2 to add data to the message tag's empty row.

- Message Tag - After the user enters Message Tags in the list control, he needs to select the relevant Message tag which needs to carry the message to the relevant Status or Salvo Button control in the player form.
- Status - This is the actual message which would be displayed in the selected control. It can be constant if a static message is to be displayed every time or can be a property. Parameter value of an object, a Variable, a cell from a data source, or n event param from the scene objects.
- Button Message Tag - It is enabled when Salvo Checkbox is checked. Select the Message tag for the salvo button to define the message for the salvo button.

Button Status: Sets the status on the Salvo Button. It can be constant if a static message is to be displayed every time or can be a property. Parameter value of an object, a Variable, a cell from a data source, or n event param from the scene objects.

- Button Status -
- Toggle Color - Sets the color of the button when pressed for 2nd action. It can be constant if a static color is to be set or can be a property. Parameter value of an object, a Variable, a cell from a data source, or n event param from the scene objects. These messages can be anything like Scene Paused, Scene Playing, Data Received, Data Updated or anything relevant to the scene.
- Normal Color - Sets the color of the Button when not pressed. It can be constant if a static color is to be set or can be a property. Parameter value of an object, a Variable, a cell from a data source, or n event param from the scene objects.

Set Element Filter

Set Element Filter: is designed to filter the view to extract sequence of numbers from multiple rows of a column and provide it as one data set for the Set Element action. For e.g. To extract the sequence of Seats won by a party from a list of constituencies. The extracted data is added to a variable selected in the Target of the action. The target can be a variable or a Tag where this data would be added and later can be used in the Set Element action to be applied to the respective AI or an FBX.

It has two Data Points:

Target: Expands a list of Objects or Variables to select the relevant Tag or Variable to Put the filtered data.

Table: Opens the list of table and allows user to specify the column from which the data is to be collected. User is required to select the 1st cell of the column to provide the column.

The Text box below displays the data set which is captured from the selected column. Note the Numbering is set from the data numbering -1 as the Set Element is done based on numbering starting from 0.

Set Variable

Is used to set the value of any variable.

- Target- Is used to select Variable or Data Source as primary value to select the source to change.
 - Variable- Is used if a User Defined, System and Global variable needs to be selected as source variable.
 - Datasource- Is used to select a cell of UDT, SQL, XML or Excel addin tables as the source variable.
- Constant- Is used to set a constant value to the selected Target Variable or Data Source.
- Variable- Is used to define an Object, Variable, DataSource or Event Param(changes according to the selected event type) from which the value will be assigned to Target Variable or Data Source.

String Format

String Format Action - Use String Format if you need to insert the value of an object variable into another string. User can define the required pattern to display the calculated value.

For e.g. the temperature is {0} •C.

Here alias {0} is a value derived from an object or variable or data table and added to a string to make a meaningful representation.

Objects | Addins: Shows the list of Objects and Variables present in the scene, which can be used as Aliases to define the pattern.

Aliases: Displays the list of objects added as Aliases.

Format: Is a text area in which the user would define the required format

Update UDT Row

Sets the value specified by user in the specific row of the UDT. User can set Status like Active/Playing or empty based on which row of UDT is currently used to provide data to the current. This is most useful with graphics which do not send status updates to the UDT for e.g. Scroller graphic where regular updates are required for data entry operator to know which row is being played and active On-Air.

- Status: Provides option to set the Status Column of the UDT. User can Set Active or Playing status based on the row being used to provide data in the graphic.
- Increment by: Specifies the increment value by which the counter column of the UDT would get updated when a particular Row is played.

Named Event

Raises an event by the name defined by the user. This event can be captured in the same or other scenes to trigger an action or a set of actions. User can select a Constant, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables. The event can also be raised on selected levels as well.

Menu Event

Is used to create context menu for Waspi.

- Tools– Enables selection of different shapes that can be drawn in Waspi environment.
- Property– Enables selection of various properties within the tools to add more flexibility to them.
- Source– Enables selection of the object on top of which the Menu Events will execute the creation of the tool using the properties defined.
- Action Tool– Lists all the tools created in the Menu Event.
- Action Stack– Lists all the actions at that keyframe.
- Use Expression Builder– Enables user to write set of expressions to build up an executable command/customized tool.
- Use default value– Rather than using the expression builder the user can also opt for default values.

Next Graphics

It triggers the next instance in the sting client rundown to Load (if the player is empty) or Play (if the player is loaded) on the selected Sting Server.

Play Action Set

Is used to execute an action set. User can add multiple play action sets to the action stack if required.

Constant – Constant displays list of actions sets in the scene.

Variable – Variables allows user to define a variable to select action set. It is helpful when the user needs to assign different actions sets in scene at runtime or based on a code.

Data Source - allows user to select action set from a cell of UDT, SQL, XML or Excel addin tables. It is helpful when the user needs to assign different animation sets from the scene at runtime or based on a code.

Raise Event

It raises an event in the scene. This event can be captured in the same or other scenes to trigger an action or a set of actions.

Raise Trigger

It raises a trigger. Custom triggers range within 0 – 1000 with a unique Trigger ID. A Trigger can be raised at various levels.

- Internal– Is raised and received in the same template.
- Template– Is received on the player of same template which has raised the trigger. This trigger can be used by coding in Comb Builder.
- Playlist– Is received on all loaded player(s) of a playlist in which the player of raising trigger is present. This trigger can be used by coding in Comb Builder.
- Sting Client– Is received on all loaded player(s) of all opened playlists of a Sting Client in which the player of raising trigger is present. This trigger can be used by coding in Comb Builder.
- Broadcast– Is received on all loaded player(s) of all opened playlist of all Sting Client(s) connected to the same Kernel Controller. This trigger can be used by coding in Comb Builder.
- Sting Server (Active Scene)- Is received on all the scenes being rendered as on the same server.
- **Sting Server (All)** - Is received on all the scenes which are loaded on the same server.
Level states the stage to which the trigger will bubble-up. These triggers can be captured in the same or other scenes to trigger an action or a set of actions.
- GPI– Is used to raise or receive trigger from a GPI device connected to the output engine. Various ports can be used to raise different triggers.

Set Current Tool

Action is used to set the tool for Sting Server based touch gesture based actions. User can perform touch operations to quickly modify object positions, rotations, scale etc. directly in Sting Server window. This tool also enables user to Draw using Freehand, Circle Rectangle etc. or create 3D objects as per

requirement. This helps to eliminate the user to create WASPi scenes for Mimosa and directly do Multi-Touch interaction in the Sting Server playout window.

User can select the relevant tool and modify the parameters as per requirement to make them function directly in the Sting Server window

Motion Trail

- Motion Trail: This effect creates a trail of object(s) behind the animated object along its trajectory. Length can be defined using the Motion Steps and Motion Frame properties and fade of trail can be defined using the Trail Transparency graph.
 - **Target** – allows user to select the object on which control must be applied.
 - Motion Trail- enables or disables the Motion Trail property.
 - Motion Steps- Step count applied on the object.
 - Motion Frame- are the number of copies to be created of the target object.

GoTo

Is used to jump to a frame of the scene, object or its tracks. User can define the Target object or Animation Set applied to the object to jump to a specified frame. The number of frames can be assigned as a constant, variable or the data source.

1. **Target**- Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.

2. **AnimSet**- When checked, the GoTo action works on the selected Animation Set.

Note: The selected Animation Set should have been already applied on the target node's track for this GoTo action to work.

3. **Frame**- defines the destination frame number for GoTo command. User can select a Constant, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables. The Frame value also works in negative if relative is turned on else it only takes positive values.
4. **Relative**- Allows user to set a relative value of leap-back/forward in respect to the existing keyframe's frame value

For e.g. Sets The relative key frame value can be set a negative value, so if the user requires to loop an animation, he can provide a value like -25 in GoTo so even when the animation keyframes are moved, the control would still work and no change in the frame value is required.

Jump

This method is used to animate/move the menu group instances. This control is usable with Menu groups or scrub-able objects.

1. **Target** – Allows user to select the object on which control must be applied.
2. Play Steps- Is the number of frames, by which animation will Jump.
3. **Block Size** – Is the speed with which menu group or object will animate. A 0 value will snap the menu group to the frame number defined in play steps, and 1 will animate the menu group or object at regular speed and any number greater than 1 will multiply the speed of animation of the menu group or object.

Pause

Is used to pause the timeline of the scene, object, animation tracks or the animation set applied on the object at a frame. The Pause can be Infinite (which can be released manually or by a trigger/event in the timeline or action set) or Time based (which releases automatically after the time specified in Hours, Minutes or seconds).

1. **Target**- Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
2. **AnimSet**- When checked, the Pause action works on the selected Animation Set.
Note: The selected Animation Set should have been already applied on the target node's track for this Pause action to work.
3. Infinite/Constant/Variable- Defines the duration of Pause in Hours (hh): Minutes (mm): Seconds (ss) or Infinite. User can select a Constant, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables.

Play Ticker

Is used to triggers the Ticker to play Ticker Scene in the sting client playlist. Commonly used with category based tickers.

Play

Is used to play the timeline of the scene, object, animation tracks or the animation sets applied on the object from a frame or release the timeline from Pause state.

1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.

2. **AnimSet** – When checked, the Play action works on the selected Animation Set.

Note: The selected Animation Set should have been already applied on the target node's track for this Play action to work.

3. **From** – Allows user to execute Play command from a specified frame number. User can define a Constant, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables. The Frame value also works in negative value if relative is turned on else it only takes positive values.
4. **Relative** – Allows user to set a relative value of leap-back/forward in respect to the existing keyframe's frame value keyframe.

For e.g. The relative keyframe value can be set to a negative value, so if the user requires to loop an animation, he can provide a value like -25 in From option so even when the animation keyframes are moved, the control would still work and no change in the frame value is required.

Reverse

It inverts the play direction of the timeline of the scene, object, animation tracks or the animation set applied to an object. If the timeline is playing forward, it starts to play in backward direction and vice-versa.

1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
2. **AnimSet** – When checked, the Reverse action works on the selected Animation Set.

Note: The selected Animation Set should have been already applied on the target node's track for this Reverse action to work.

Change Key Frame

Is used to change the value of a Key frame of an object's translation/parameters or an animation set.

1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
2. **Frame** – Displays the list of Key Frames on the selected track.
3. **Relative** – When turned on it adds to the exiting values of the key frame. Turn off if defining an absolute value of the object's key frame value.
4. **Constant/Variable** – Defines the new value for the selected Key Frame. User can define a Constant, Object, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables.

Clear Buffer

The Clear Buffer adion resets the contents of a specified rendering buffer (such as a canvas or render texture), typically by filling it with a uniform color as specified in the Clear Color box or clearing its data. This process prepares the buffer for new rendering operations, ensuring that previous content does not interfere with the upcoming frame, and is used to maintain a clean and consistent rendering surface or Canvas.

Buffer - Displays the list of available Canvas or Render Texture nodes,

Clear - When checked, allows user to specify a clear color for the selected node.

ZBuffer - When checked, allows user to clear the Canvas at the specified Z Order

Mode

This method is used to change the type of animation from Play mode (Playout animation) to Step (Scrubable) animation. This mode is used in Touch based scenes.

1. **Target** – Allows user to select the object on which the animation Mode will be changed.
2. **Mode** – Allows user to select the Animation mode – Play/Step.

Move Key Frame

Moves keys of a desired targeted object to a desired frame/location on the timeline, when this command is received during scene play.

1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
2. **Frame** – Displays the list of Control Keys and Key Frames on the selected track.
 - **Control Key** – Option is used when user wants to move a control key
 - **Keyframe** – Option is used to move the keyframe
 - Frame No. - Displays the list of Keyframes and/or Control Keys of the respective track as absolute frame numbers.
 - **Index** – Displays the list of Keyframes and/or Control Keys of the respective track as serial no. in order they appear on the track.
 - **Attach Keyframe** – This option appears when Control Key option is selected in Frame Menu. It helps users to link the Keyframe of the selected Frame or Index with the Control Key which would also get moved when the control key is moved to new specified track key value.

3. **Relative** – Allows user to set a relative value of leap-back/forward in respect to the existing Keyframe or Control Key frame.

For e.g. The Keyframe is at frame 20 and has a control key like GoTo, the user can move the keyframe to 10 by specifying -10 in the constant value or 30 if its 10.

4. **Constant/Variable** – Defines the new value for the selected Keyframe or Control Key. User can define a Constant, Object, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables. Excepts negative value when relative is turned on.

Page In

This command sends the Page-in trigger to the Scene. Page-in is commonly used with Page Controller object. It Renders ON the Page Controller Group and Plays it from Frame '0' by default.

Target – Allows user to select the Page Controller Object.

Page Out

This command sends the Page-out trigger to the Scene. Page-out is commonly used with Page Controller object. It checks if there are no more pages in the Data Pages controller and if Loop command is turned off, it will exit from the page animation loop and trigger the Default Controller.

Target – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.

Play Complete

This command sends the Play Complete trigger to the Scene. When applied on the SGViewAll track – if scene raise this trigger at Sting-Client/Server it unloads itself from the player.

Render

Enables or disable the rendering of the target object from the scene.

- Target – Selects the object to Render On/Off
- Render – Sets the render state true or false to be defined as Constant or based on a Variable or an Object's parameters. Any value other than 0 is considered true and will result in Render True. Empty or 0 value will result in render False. User can also select String based parameters to determine the same. Empty or 0 value in string will denote false.

Scene

- Mode: this method is used to change the type of animation from Play mode (Playout animation) to Step (scrubbable) animation. This mode is used in Touch based scenes.
 1. **Target** – Allows user to select the object on which the animation Mode will be changed.
 2. **Mode** – Allows user to select the Animation mode – Play/Step.
- Page In: This command sends the Page-in trigger to the Scene. Page-in is commonly used with Page Controller object. It Renders ON the Page Controller Group and Plays it from Frame '0' by default.
 1. **Target** – Allows user to select the Page Controller Object.
- Page Out: This command sends the Page-out trigger to the Scene. Page-out is commonly used with Page Controller object. It checks if there are no more pages in the Data Pages controller and if Loop command is turned off, it will exit from the page animation loop and trigger the Default Controller.
 1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
- Play Complete: This command sends the Play Complete trigger to the Scene. When applied on the SGViewAll track – if scene raise this trigger at Sting-Client/Server it unloads itself from the player.
- Render: Enables or disable the rendering of the target object from the scene.
 1. **Target** – Selects the object to Render On/Off
 2. **Render** – Sets the render state true or false to be defined as Constant or based on a Variable or an Object's parameters. Any value other than 0 is considered true and will result in Render True. Empty or 0 value will result in render False. User can also select String based parameters to determine the same. Empty or 0 value in string will denote false.
- Set Material: it applies the selected material on the Target object.
 1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
 2. **Constant** – Displays list of materials in the Material Pool.
 3. **Variable** – Is used to select a User Defined, Server or System variables which can be used as a source in the expression.
- Set Parent: Allows swapping the group of the target object to a new group.
 1. **Target** – Selects the object to be moved to new group.
 2. **Inherit Parent Transform** – when checked, appends the transformation of the parent object to the child object.
 3. **Target** – Selects the new group to which the Target object will be moved.
- Set Pivot: Sets the pivot of any object at any targeted frame number.

1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
 2. **The Cubical Figure** – Click on the points in the figure to set the pivot.
 3. **Pivot**– Set pivot point by using combination of the buttons listed for each axis.
 4. **Custom Pivot** – Enter the value for pivot points manually for each axis.
- Change Key Frame: Is used to change the value of a Key frame of an object's translation/parameters or an animation set.
 1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
 2. **Frame** – Displays the list of Key Frames on the selected track.
 3. **Relative** – When turned on it adds to the exiting values of the key frame. Turn off if defining an absolute value of the object's key frame value.
 4. **Constant/Variable** – Defines the new value for the selected Key Frame. User can define a Constant, Object, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables.
 - Move Key Frame: Moves keys of a desired targeted object to a desired frame/location on the timeline, when this command is received during scene play.
 1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
 2. **Frame** – Displays the list of Control Keys and Key Frames on the selected track.
 - **Control Key** – Option is used when user wants to move a control key
 - **Keyframe** – Option is used to move the keyframe
 - Frame No. - Displays the list of Keyframes and/or Control Keys of the respective track as absolute frame numbers.
 - **Index** – Displays the list of Keyframes and/or Control Keys of the respective track as serial no. in order they appear on the track.
 - **Attach Keyframe** – This option appears when Control Key option is selected in Frame Menu. It helps users to link the Keyframe of the selected Frame or Index with the Control Key which would also get moved when the control key is moved to new specified track key value.
 3. **Relative** – Allows user to set a relative value of leap-back/forward in respect to the existing Keyframe or Control Key frame.

`For e.g. The Keyframe is at frame 20 and has a control key like GoTo, the user can move the keyframe to 10 by specifying -10 in the constant value or 30 if its 10.`
 4. **Constant/Variable** – Defines the new value for the selected Keyframe or Control Key. User can define a Constant, Object, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables. Excepts negative value when relative is turned on.

- Wire: It wires keyframe/track or non-interpolated parameters of an object with the given data.

1. **Target** – Selects the object's animation track to wire.
2. **Frame Wire** – When checked, will allow to select the keyframe to be wired of the target object.
3. **Index** – Displays the key frame in a sequence of 1, 2, 3 simultaneously for frames say 10, 20, 30
4. **Wire** – When Checked, will wire the Track/Key of the selected object and vice versa when unchecked.
5. **Constant/Variable** – Defines the new value for the selected Key Frame. User can define a Constant, Object, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables.

- Set Param: defines the parameter values of the specified object on runtime.

User can define a Constant, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables.

1. **Target** – allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
2. **Constant/Variable** – defines the new value for the selected Key Frame. User can define a Constant, Object, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables.

- Set Element: Enables setting the color properties of objects like AI (Polydata) and FBX (materials).

1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
 - **All** – Applies the specified value on all the Indexes of the AI or FBX.
 - **Range** – Specifies the range (for e.g. 1-10) or List (for e.g. 1,3,6,8,9,15 etc.) of Polydata or FBX indexes to be applied with the specified Value.
 - **Value** – Specifies the Color Value for Polydata indexes or Material for FBX objects. Only the selected objects based on All or Range will be applied the specified Value.

- **Set Sampler** – This action applies the specified texture to a material's texture stage. User can apply texture to any one or all texture stages at once using this option.

1. Target: Allows user to select the material to which the texture needs to be applied.
 - Stage 1/2/3/4- Applies the texture to the Stage 1/2/3/4 of the selected material respectively.
 - Constant: Manually enter the name of the texture available in the texture pool.
 - Object: Select the texture from the list of available textures in the texture pool.
 - Variables: Select the variable which will specify the name of the material.
 - Data Source: Select the variable which will specify the name of the material.
 - Event PARAM: Select the event PARAM which will specify the name of the material.

- Set View Param action is created to control the Drone Designer View objects.

View objects are an advanced form of Render Texture which can be called between scenes and each view can have a separate camera to view the selected objects of that respective view.

User can add all scene objects or selected scene objects to a View and can render it in other scenes. (The scene which must be rendered in the host scene is required to be loaded as well). By default, the loaded scene gets loaded in the default view.

User can hide the default view or any view in the view tab, if not required to be rendered in the scene. Set View Param action is also used to control the above mentioned features during scene runtime and can also be defined by using Salvo or Action Set Button.

1. Target: Sets the view for which the parameters are required to be set.

Camera: Sets the Camera from which the selected view should be displayed. It can be constant or can be a property. Parameter value of an object, a Variable, a cell from a data source, or n event param from the scene objects.

2. **ViewIndex** – Sets the index level at which the view should render. There are limited number of view indexes so user can select to render multiple views on the same view index also. It can be placed at different locations using the Viewport position property.
3. **Render** – Provides a checkbox to enable or disable the rendering of the selected view in Target.
4. **Viewport** – Sets the position and size of the view to be rendered on the selected View Index. User can render multiple Views on same index using Viewport Position and Scale Parameters. Both the parameters are based on Percentage from 0-100. User can define 0,0 and 50, 50 if required to position the view on the top left corner. in this case, the 0,0 is the starting position in X and Y and 50, 50 is half of the viewport(if HD it would be 960x540px and if SD it would be 360x288px).
5. **Tag** – Is a string property to put description of the View or can also be treated as a variable to associate View with certain objects. For e.g. User can set Camera name as Tag, and In the Camera, Property take the name of the Camera from the view's Tag property. This results in faster copying of controls as the Tag property would be common for all views and the relative free camera would be automatically taken from the Tag property.
6. **Change Sequence** – This enables user to set the render order of the Views. The View on sequence 1 would be rendered 1st and so on. In case user wants object of a View 3 in View 1. The View 3 must be rendered 1st to be made available for View 1. So, user can order View 3 on top of View 1. User select and drag to order the views in the Change Sequence list.

- Clear Buffer

Set Element

Enables setting the color properties of objects like AI (Polydata) and FBX (materials).

- **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
 - **All** – Applies the specified value on all the Indexes of the AI or FBX.
 - **Range** – Specifies the range (for e.g. 1-10) or List (for e.g. 1,3,6,8,9,15 etc.) of Polydata or FBX indexes to be applied with the specified Value.
 - **Value** – Specifies the Color Value for Polydata indexes or Material for FBX objects. Only the selected objects based on All or Range will be applied the specified Value.

Set Material

It applies the selected material on the Target object.

1. **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
2. **Constant** – Displays list of materials in the Material Pool.
3. **Variable** – Is used to select a User Defined, Server or System variables which can be used as a source in the expression.

Set Param

Defines the parameter values of the specified object on runtime.

- **User can define a Constant, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables.**
1. **Target** – allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
 2. **Constant/Variable** – defines the new value for the selected Key Frame. User can define a Constant, Object, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables.

Set Parent

Allows swapping the group of the target object to a new group.

1. **Target** – Selects the object to be moved to new group.
2. **Inherit Parent Transform** – when checked, appends the transformation of the parent object to the child object.
3. **Target** – Selects the new group to which the Target object will be moved.

Set Pivot

Sets the pivot of any object at any targeted frame number.

- Target – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
- The Cubical Figure – Click on the points in the figure to set the pivot.
- Pivot– Set pivot point by using combination of the buttons listed for each axis.
- Custom Pivot – Enter the value for pivot points manually for each axis.

Set Sampler

This action applies the specified texture to a material's texture stage. User can apply texture to any one or all texture stages at once using this option.

1. Target: Allows user to select the material to which the texture needs to be applied.
 - Stage 1/2/3/4- Applies the texture to the Stage 1/2/3/4 of the selected material respectively.
 - Constant: Manually enter the name of the texture available in the texture pool.
 - Object: Select the texture from the list of available textures in the texture pool.
 - Variables: Select the variable which will specify the name of the material.
 - Data Source: Select the variable which will specify the name of the material.
 - Event PARAM: Select the event PARAM which will specify the name of the material.

Set View Param

Action is created to control the Drone Designer View objects. View objects are an advanced form of Render Texture which can be called between scenes and each view can have a separate camera to view the selected objects of that respective view.

User can add all scene objects or selected scene objects to a View and can render it in other scenes. (The scene which must be rendered in the host scene is required to be loaded as well). By default, the loaded scene gets loaded in the default view.

User can hide the default view or any view in the view tab, if not required to be rendered in the scene. Set View Param action is also used to control the above mentioned features during scene runtime and can also be defined by using Salvo or Action Set Button.

1. Target: Sets the view for which the parameters are required to be set.
Camera: Sets the Camera from which the selected view should be displayed. It can be constant or can be a property. Parameter value of an object, a Variable, a cell from a data source, or n event param from the scene objects.

2. **ViewIndex** – Sets the index level at which the view should render. There are limited number of view indexes so user can select to render multiple views on the same view index also. It can be placed at different locations using the Viewport position property.
3. **Render** – Provides a checkbox to enable or disable the rendering of the selected view in Target.
4. **Viewport** – Sets the position and size of the view to be rendered on the selected View Index. User can render multiple Views on same index using Viewport Position and Scale Parameters. Both the parameters are based on Percentage from 0-100. User can define 0,0 and 50, 50 if required to position the view on the top left corner. in this case, the 0,0 is the starting position in X and Y and 50, 50 is half of the viewport(if HD it would be 960x540px and if SD it would be 360x288px).
5. **Tag** – Is a string property to put description of the View or can also be treated as a variable to associate View with certain objects. For e.g. User can set Camera name as Tag, and In the Camera, Property take the name of the Camera from the view's Tag property. This results in faster copying of controls as the Tag property would be common for all views and the relative free camera would be automatically taken from the Tag property.
6. **Change Sequence** – This enables user to set the render order of the Views. The View on sequence 1 would be rendered 1st and so on. In case user wants object of a View 3 in View 1. The View 3 must be rendered 1st to be made available for View 1. So, user can order View 3 on top of View 1. User select and drag to order the views in the Change Sequence list.

Wire

It wires keyframe/track or non-interpolated parameters of an object with the given data.

1. **Target** – Selects the object's animation track to wire.
2. **Frame Wire** – When checked, will allow to select the keyframe to be wired of the target object.
3. **Index** – Displays the key frame in a sequence of 1, 2, 3 simultaneously for frames say 10, 20, 30
4. **Wire** – When Checked, will wire the Track/Key of the selected object and vice versa when unchecked.
5. **Constant/Variable** – Defines the new value for the selected Key Frame. User can define a Constant, Object, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables.

Set Scroller

User can add/remove scroller group from scroller object hierarchy through this action. There are three options for the selected scroller implementation

- Insert Next: Inserts the new scroller after the current scroller group tail is entered in the scroller area.
- Insert Now: Inserts the new scroller just at the current position of the scroller
- Remove (Abort): Removes the scroller group from the scroller set after the tail exits from the scroll area. When Abort is checked, it removes the scroll group immediately. Next scroller group moves to fill the emptied position.

Start Scroller

It plays ScrollerNode object in the scene.

- Target – allows to select the scroller node to start.

Stop Scroller

It stops ScrollerNode object in the scene.

- Target – allows to select the scroller node to start.
- Abort – if checked will stop the scroller node and unload it, if unchecked it will wait for the last instance in the scroller to scroll out.

Timer/Text Scroll

Is used to Start, Pause or Stop the Countdown clock or Stopwatch.

- Target – Allows user to select the 3D Text object. Target object should be a 3D or 2D Text in Countdown or Stopwatch mode.
- Start – Starts the Stopwatch Count.
- Pause – Pauses the Stopwatch Count.
- Stop – Stops the Stopwatch Count.

Play

Plays the target animation on the target 3d Skeletal Mesh object.

- **Target** – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
 1. **Motions** – Displays the list of Skeletal Animation Sets available in the Scene.
 2. **Start Frame** – Defines the Starting Frame of the skeletal animation set.
 3. **Stop Frame** – Defines the End Frame of the skeletal animation set.
 4. **Speed** – Defines the speed of animation of the Skeletal 3D Mesh. Default value 1 is the normal speed of animation, increase or decrease the value to increase or decrease the speed of animation.
 5. **Char Loop** – Defines the number of times to loop the animation.
 6. **Blend Frame** – Defines the number of frames to blend between two different animations applied on the same 3D Skeletal Mesh.
 7. **Transition Time** – Defines the time frame after which another animation should start playing.

8. **Track Type** – Provides an option to select the animation type to classify the same as Facial or Body animation. If the user wants to apply 2 different animations to work simultaneously on the 3D Object, he can select Track Type Body in one and Facial in other.

Note: Remove previously applied AnimSet before applying it again as it may lead to unusual behavior.

Stop

Stops the target animation on the target 3d Skeletal Mesh object.

- Target – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
- Motions – Displays the list of Skeletal Animation Sets available in the Scene.
- Retain Transform – Enables or disables, whether the Model should remain in its current pose or return to its pose that was before applying the AnimSet.
- Track Type – Provides an option to select the animation type to classify the same as Facial or Body animation. If the user wants to apply 2 different animations to work simultaneously on the 3D Object, he can select Track Type Body in one and Facial in other.

SetCrossPoint

This command enables user to switch the input and output ports of the Video Router. This command is usable in combination with Router Control addin.

- Addin – Allows user to select the Router Addin for the specific switcher.
- Input Port – Allows user to select a Constant, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables to select the input port of the Video Router.
- Output Port – Allows user to select a Constant, Variable or a Value from selected cell of UDT, SQL, XML or Excel addin tables to select the output port of the Video Router.

Render Target

It maps 3D output to specify surface.

Blur Texture

This texture type can be used for real time implementation of Depth of Field (DOF) or for the real time implementation of Screen Space Ambient Occlusion (SSOA).

Time Code

Update Buffer

Allows user to update the Addin data on demand. This action is only required if the Update Buffer checkbox is disabled in the Data Addin.

Pause Video

It pauses the selected audio or video.

Target – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.

Stop Video

Stop Video action allows user to stop the playing video. It has three options.

- Reset – Resets the video to frame 0
- Abort – Aborts the video play and makes the current frame blank and shows the material diffuse color
- Current – Stops the video at the currently playing frame. It works same as Pause action.

Play Video

Starts playing the selected audio or video.

- Target – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.
- Loop – Enables or Disables the Video loop after it reaches the last frame.
- Speed – Defines the play speed of video. Default value 1 is the normal speed of video, increase or decrease the value to increase or decrease the play speed of target video.

Load Scene

Loads a scene in the Engine. It requires a WScene Variable to select the scene based on instance id or scene id.

- Variable: Acquires the name of the scene to load from the specified variable.
- Playlist Instance: Choice to load the Waspi template from the Playlist by Instance Slug or by Name of Template
- Instance Slug: Select to load the scene from sting client playlist by instance name.
- Template Name: Select to load the scene from sting client playlist by template name.
- Local File: Select to load the scene by specifying the file-path of the scene located in the local drive.
- SLUG: Specifies the Instance/Template name or Local file path of the scene to be loaded.
- ZORDER: Specifies the layer-order of the scene to be loaded. 0 is for background.

Play Scene

Plays the loaded scene in the engine.

Target: Acquires the name of the scene to load from the specified variable.

Action Stack: List the set of actions to be performed while playing the scene.

Render Scene

Renders on/off the loaded scene in Waspi Mimosa.

- Target: Acquires the name of the scene to render on/off from the specified variable.
- Action Stack: List the set of actions to be performed while rendering the scene.

Save Scene

Scene Cue

Cues the template loaded in the WScene Variable used in Load Scene.

- Default: Cues Self
- Variable: Provides option to Cue any other template as specified in the WScene variable used with Load Scene action.

Unload Scene

Unloads the loaded scene from engine.

- Target: acquires the name of the scene to render on/off from the specified WScene variable.

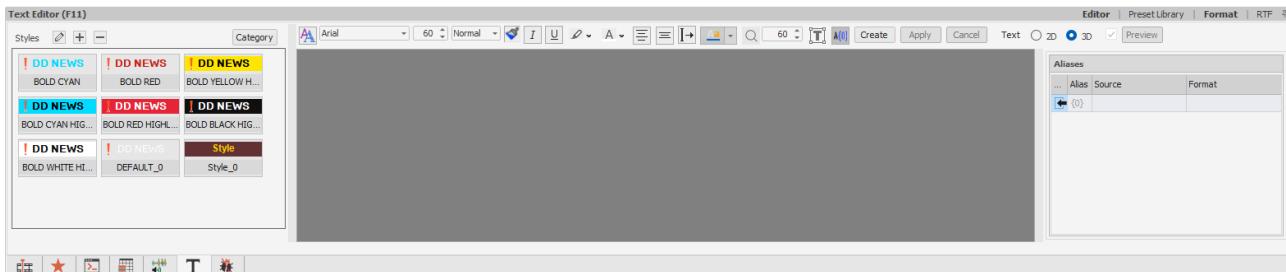
- Action Stack: list the set of actions to be performed while unloading the scene.

Wasp Scene

- Load Scene: Loads a scene in the Engine. It requires a WScene Variable to select the scene based on instance id or scene id.
 - Variable: Acquires the name of the scene to load from the specified variable.
 - Playlist Instance: Choice to load the Waspi template from the Playlist by Instance Slug or by Name of Template
 - Instance Slug: Select to load the scene from sting client playlist by instance name.
 - Template Name: Select to load the scene from sting client playlist by template name.
 - Local File: Select to load the scene by specifying the file-path of the scene located in the local drive.
 - SLUG: Specifies the Instance/Template name or Local file path of the scene to be loaded.
 - ZORDER: Specifies the layer-order of the scene to be loaded. 0 is for background.
- Play Scene : Plays the loaded scene in the engine.
 - Target: Acquires the name of the scene to load from the specified variable.
 - Action Stack: List the set of actions to be performed while playing the scene.
- Render Scene: Renders on/off the loaded scene in Waspi Mimosa.
 - Target: Acquires the name of the scene to render on/off from the specified variable.
 - Action Stack: List the set of actions to be performed while rendering the scene.
- Unload Scene: Unloads the loaded scene from engine.
 - Target: acquires the name of the scene to render on/off from the specified WScene variable.
 - Action Stack: list the set of actions to be performed while unloading the scene.
- Scene Cue: Cues the template loaded in the WScene Variable used in Load Scene.
 - Default: Cues Self
 - Variable: Provides option to Cue any other template as specified in the WScene variable used with Load Scene action.
- Save Scene

Text Editor (F11)

3D and 2D Text – Text Editor now available as an exclusive panel and allow faster access as well as on the go edit of text. Text Style and Presets are added to the Text Editor.



1. Styles have been added to 3D Text. Use a combination different parameter like bold, italics, color font etc. to make one style of text. Each style can be associated in a category and later use by journalists to apply to rich text in data entry forms. User can save multiple styles of text in a category. And multiple categories can be created to accommodate the respected styles.

Text Panel allows the write and apply the text and create 2D or 3D Text with the written text directly from the Text Panel. This allows users to write or paste the text from any source and create the text directly in the viewport. More properties added in the interface allow most of the text edit properties to change from the same panel. Other less used properties can be changed from the parameters panel.

Styles in the data entry form can be applied to Rich-text Box controls. User can select the Style Bar parameter in the property window and select which category of styles he wants the journalist to see in the Rich Text box when he wants to enter data. Style option only provides a toolbar with the styles of the selected category. Font provides the complete toolbar having the font, size, color, bold, italics etc. properties for Rich Text.

User can opt to show both or any one of them to enable changes in the text style by the journalist for data entry. User can drag and drop styles in the category to required position.

2. Auto Preview of the selected text changes in the viewport.
3. Font Stretches will be displayed along with font names in the Font Combo box.
4. Superscript and Subscript option is added. Select the Text, Right-Click Context Menu > Format > select the relevant option.
5. Small Caps format has been added.
6. Digit Substitution has been added to work with Date and Time formats. This helps the user to display the digit in Date and Time in the context specified from the list of Cultures in the format window.
7. Text Highlight & Underline option has been added to RTF text.
8. Scroll option added to Action on Overflow parameter in Drone Designer. This option helps users to auto-scroll the excess text in bounded text. The text also has a typewriter effect on the last line of the scrolling text. This option is useful with long paragraph-based text where the content is more but does not require the paging functionality.

9. Text Manipulator interface has been updated. Now all the parameters are changeable from the render viewport only. This increases efficiency and a one-place text edit for the selected string. Click on the Font to get a combo box to change the font, Click on Text weight to update the text weight like Bold, Normal etc. Click on "I" button to italicize the text. Click on the Alignment button to horizontally align the text, Click and Drag on Text size and Extrude numbers to change the size and extrude of the selected text.
10. Bounded box is re-sizeable in viewport using the Manipulator Gizmo. It gets is enabled in Manipulator mode.
11. **Font combo box** – Five most recent Fonts used in design are added at the top of list of Fonts with a separator.
12. Variable Format option has been added to Text3D and Text2D object. This allows user to concatenate variables and with the string. This helps user to reduce the number of text objects as well as wiring to show multiple texts in one string. For e.g., 36°C where the value 36 is coming from a weather feed (variable in XML Add-in) and °C is added in the format string. Similarly, multiple variable usage to combine and show a financial data for an Index.
13. Word and Line animation modes have been added to text. Previously only Character wise animation was possible.

Presets in Text allow user to add animation to one or multiple 3D Text objects. User can apply from a variety of presets available in the interface and These animations are made using parameters so if any Scale or Rotation is already applied to the object, which would need to be reset to defaults as the animation of presets can look different because of transformation changes. However, translation can changes do not matter as the base key can start from that value.

RTF Editor allows user to create and edit the 3D and 2D rich-text objects in a scene. User can press the 'F11' function key to focus the Text Editor window in the Drone Designer UI. RTF allows user to add styles and String Formats. Rich Text allows user to use multiple Text Format properties like Bold, Italics, Underline etc. along with varied colors of text and highlights in the same string.

RTF editor has the 2 modes -

1. **Editor** – Allows user to create or edit the 3D or 2D text object.
2. **Preset Library** – Allows user to add animation presets to the selected 3D Text objects.

Editor or Preset mode can be selected from the right-aligned button menu located on the title bar of the RTF Editor Window.

Editor has the following sections

1. **Text Toolbar** – Contains all the Text Formatting parameters to apply Front, Color, Size, Style, Alignment or Blink styles to the selected text in the editor.
2. Text Edit box- allows user to edit the text string and also see a preview of the text styles.
3. **Styles Dock** – allows you to assign styles the selected text string.
4. **String Format Dock** – allows user to define data aliases for the text string.

Text Toolbar – User can select multiple text objects and change their parameters at one time using the text toolbar as well as Parameters dock window

-  **Style** – Show/Hide the Style dock window. The Style Window is displayed on the Left side of the RTF Editor window.

- **Font** – Displays a combo box list of system fonts to apply to the text. Default state is pressed.
-  **Italics** – Italicize the selected text.
-  **Underline** – Underlines the selected text.
- **Size** – Sets the size the selected text.
- **Weight** – Displays a list of Font weights supported by the selected font
-  **Apply Button** – Applies the previously applied Font Weight to other words or sentences in the same string.
-  **Highlight Color** – Allows user to set a text Highlight color. Highlight color is displayed as a box behind the selected text. The selected color is also displayed in the icon.
-  **Text Color** – Allows user to set the text forecolor. Text color will not display if Material Color Front is set to True. Text has 3 layers of color. Forecolor, RTF Color and Material Color. Forecolor is the bottom layer, above that is the RTF color and Material Color is the top most layer. In case there is transparency in Material Color, RTF Color/Forecolor would be displayed from the transparent area respectively.
-  **String Format** – Show/Hide the String Format Window. The String Format Window is displayed on the Right side of the RTF Editor window. Default state is pressed.
- **Text 2D/3D** – Radio button is effective when user wants to create a 3D Object based on the text and parameters currently applied in the RTF Editor Window.
- **Create** – Click on Create button to create the New 2D or 3D text as selected in the Text 2D/3D radio button.
- **Checkbox and Preview** – The Checkbox locks the preview button to and displays the changes done in the RTF Editor window in the Viewport render.
- Apply button confirms the changes applied on the selected text objects. If the apply button is not clicked, a dialog box to confirm the changes is popped when the user changes the selection of the selected Text objects. User can Apply all Changes to keep these changes or Click Cancel to cancel the applied changes to the 3D Text.
- Cancel button resets any changes done to the 3D Text object(s).

Text Edit box allows user to edit text of the selected Text Object(s). All the changes done in the parameters of the text are shown in this box. User can type text, do required formatting, select 2D or 3D text in the RTF Editor toolbar and click on Create to create a New 2D/3D text object with the selected formatting in the scene.

Styles Dock Window

 Style button is used to invoke and collapse Styles window in the RTF Editor. Style is a collection Text Parameters and it stores various combinations of text properties to make text formatting easier at design time and also convenient to keep the design theme of the channel intact.

User can create program or theme based styles and provide the access to journalist to apply designer defined text formatting in texts. Using font styles makes it easy to apply a collection of text parameters like Font, Color, Size, Bold, Italics, Highlight to a text in one time, without affecting the actual text string. For e.g. Making some words characters Bold and Highlighted in a string can be saved as a style and designer can provide access to this specific style in RTF Editor properties in the Form Designer tab.

Styles Dock window has the following features.

The Style Window shows styles of the selected category. Category can be selected using the Category button located on the right side of the Styles Dock window Title Bar.

Hover on a style to show its preview on the selected text in the Text Edit Window. Double Click on the respective thumbnail to apply the style on the selected text. A tool-tip is also shown listing the parameters included in the style.

! Icon is shown if a font in a style is missing. The name of the font is also displayed in the status bar of the style dock window. Status bar is located at the bottom of the Style dock window.

Multiple styles are collected in a Category.

- Button opens the list of Categories. Each category is like a folder for collecting various styles. User can create category to define a Program Name or a Theme and make sure that all the styles for the respective Program/Theme are stored in the respective category.
- **Category** Category window has the following features
 - Add button adds a New Category to the Category list.
 - Delete button deletes the selected category A confirmation dialog box is presented when user clicks on this button to delete the selected category.
 - Edit button opens a dialog box showing Name and Description to edit the selected category detail. User can also press F2 in the respective cell to edit the Name or Description of the category.

To add a Style – A Category must be selected to add a style. By default designer comes with some default styles. User can add more styles to these categories or create their own as required.

1. Select the formatted text based on which you want to create the style it can be a word or even an alphabet.
2. Click on the Add button in the styles window to open a dialog box.
3. Write the Name of the Style in the text box labelled Name and write the Text you want to see as Preview image in the style thumbnail.
4. User can also click and drag the icon in the style window(when the formatted text is selected) to create a style with default text Style as name and Preview Text.

String Format Dock Window



String button is used to invoke and collapse String Format window in the RTF Editor.

String Format allows user to add variables from Variable pool, Data Source tables and pre-defined constants to the text string. This helps user to maintain a single string of text and concatenate the variables into the running text string.

For e.g. If a user wants to add System time to a text string, He can add {0:h:mm tt} to the test string. In the above example, 0 is the Alias which has the Constant 'Time' added to it and 'h:mm tt' is the time format.

The String Format window has the following features

An alias is automatically created and associated with a source when it is added to the alias list. Each Alias is defined in spring '{ }' brackets and is shown in the Alias column.

Source is the address of the variable from where the data is linked. User clicks on the source window to show the list of variables. Variables are categorized in 2 tabs, Addins and Constants.

- **Addins** – show the list of Local variables, Predefined variables and Scene Variables created by the user.
- Constants - show a list of pre-defined constants which can be added to the text in various formats. Date, Time, Number, Stopwatch, Countdown type constants are available with their respective format

Format column shows a list of display variations and formats available for the selected source. Each format has the Alias number associated with it automatically for easy identification in the string. User can manually edit the format in the format column as per Microsoft string format definitions.

Format – Specify the Date or Time format as per the option selected in Text Type.

When constant is Date

```
d = day      dd= day with leading 0  
ddd = weekday name short     dddd = weekday name full  
M = month      MM = month with leading 0  
MMM = month name short      MMMM = month name full  
y = year      yy = year with leading zero    yyyy = full year
```



Note: d or dd can also be used with Countdown Time text type to show Days.

Time or Countdown Time or Stopwatch constants formats descriptions
h = hour m = minute s = second tt = AM. or PM. h/H = 12 / 24 hour
hh, mm, ss = display leading zero h,m,s = without leading zero

Number format have the following format types

```
f = float (0.00)  
c & c2 = Adds currency locale based currency symbol with two decimal places ($0.00)  
c1 = Adds currency locale based currency symbol with one decimal place ($0:0)
```



Note: Note: currency format can be defined by defining the Culture in the

Format parameters of the text.

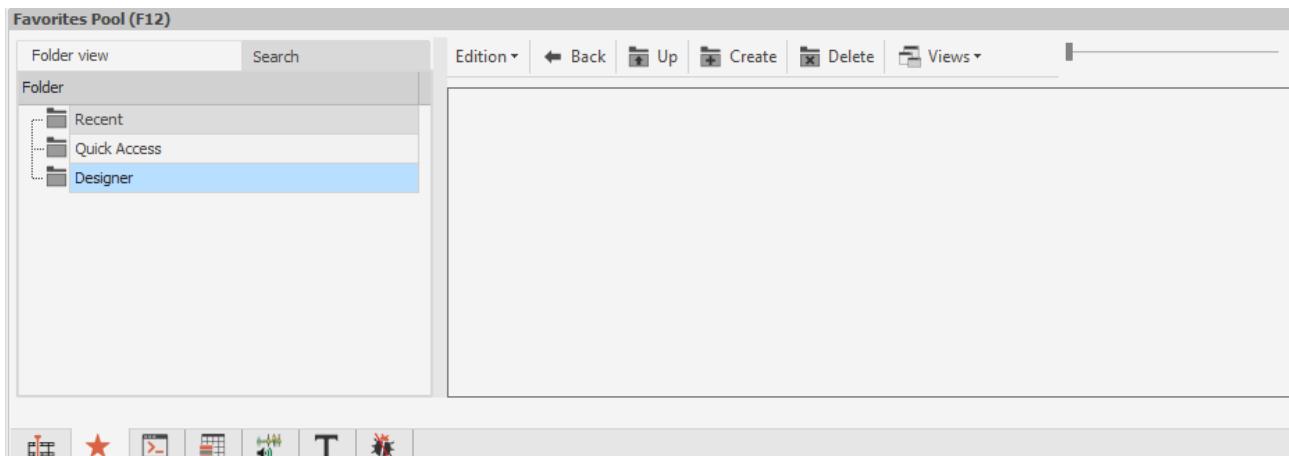
F4 = float with 4 decimal places

P = adds percentage sign to the source value (0.00%)

Favourites Pool

Favorites Pool window works as a network-shared library of commonly used assets. These network-shared assets may consist of single object, group of objects or user-specified animated tracks and all key frame information contained within.

Once the object is moved to favorites, it becomes an asset for all the users on the network and can be used multiple times in other scenes as well. The dragged object will retain all the animation, transformation, its parameters and wiring either with Addin or with other objects etc. Control keys are not copied with the object animation.



User can select an object in '**Object Pool**' or '**Hierarchy**' window and Drag it to the '**Favourites Pool**' to add it to favorites. User can select the folder before dragging the object to 'Favourites Pool' to put it in the respective folder.

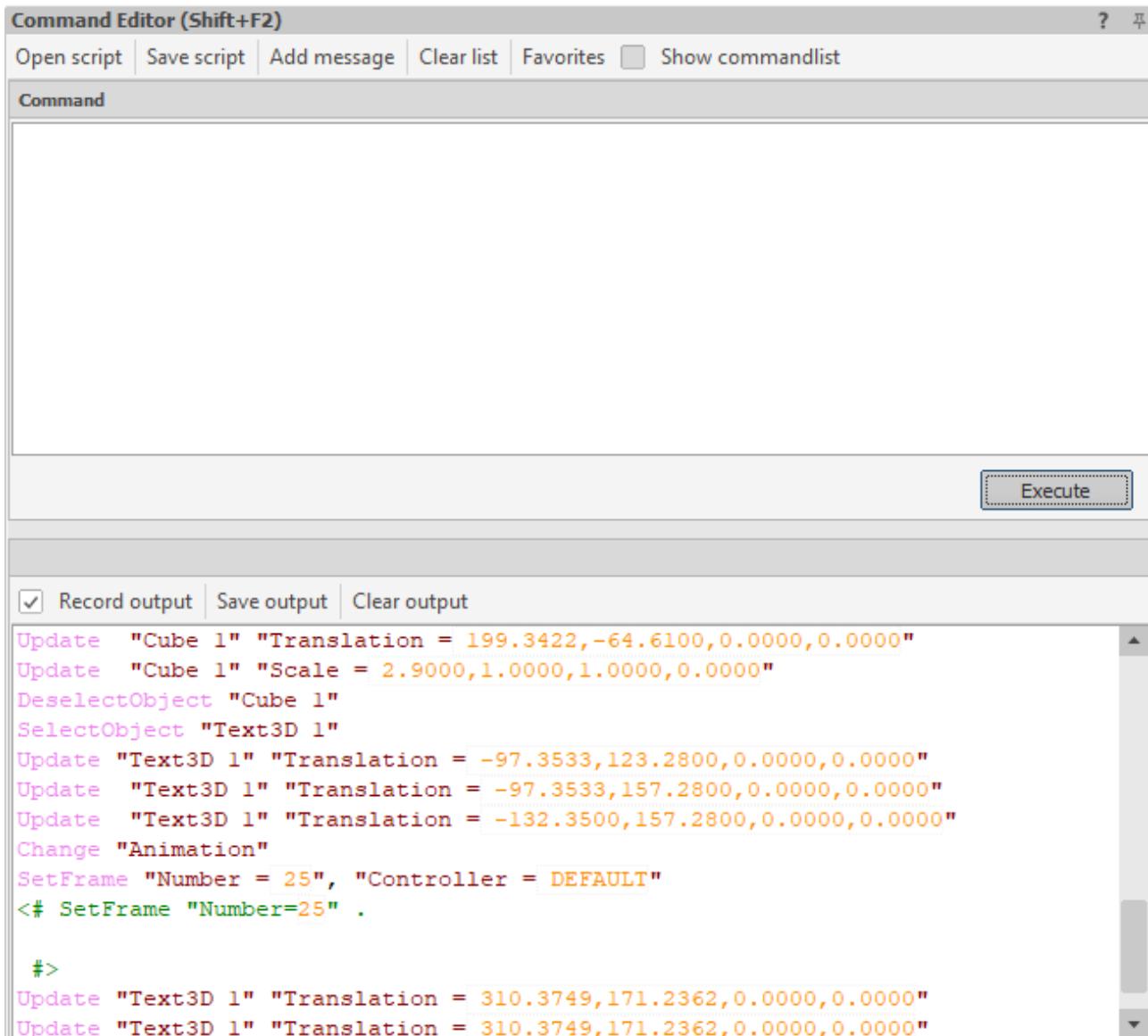
Right-click on an interpolatable track in 'Parameters' window to add the track with its animation keys to the Favorites Pool. These tracks can be dragged to object with matching parameters.

Following are the detail in reference to the illustration of Parameters window:

1. Folder View: Shows assets in its respective folder where it is saved. To list all assets at a time, select the root folder (for example, in the shown picture, Designer that contains all folders).
2. Search: Search assets by defining its search criteria including "Date Criteria," "By Keyword" and "By Creator".
3. Back: Takes to the last visited folder.
4. Up: Takes to the parent folder of the selected folder.
5. Create: Opens the Create Folder window asking folder name and its location. By default, the selected folder appears as the Current Folder.
6. Delete: Deletes the selected folder.
7. Views: Toggles the view of assets. It has two view options - 'Large Icons' and 'Details'.
 - Large Icons: Displays thumbnails of assets along with their names at the bottom.
 - Details: It displays assets in list view along with Name, Description, Size, Type, Modified, Keywords, and File Type details.

Command Editor

This interface enables the user to view the commands for activities performed in the SceneGraph. Users can also write different commands to perform specific tasks on the SceneGraph such as Create, Delete, Copy, etc.



Command Editor has 2 sections:

Editor – User can type commands and PowerShell scripts to perform these actions in Example: To create a sphere using Command Editor:

- Type "Create Sphere"
- Click the Execute button.

This way, it creates a sphere containing the default property.

The Command Editor window appears as shown in the above illustration. Following is the interface description of the Command Editor window:

- Open Script: This command is used to open the script file. Click the Open Script button, which opens a dialog box.

- Save Script: This command is used to save the script being used in the Command Editor.
- Add Message: This command is used to display message when the script is executed.
- Clear List: This command is used to clear the scripts in the Command Editor.
- Favorites: This command allows saving the scripts in Favorites list for quick and easy access.
- Show Command list: Selecting this checkbox shows the command list. By default, it remains unchecked. Selecting the Show Command list checkbox shows the command.
- Command [Input Box]: This is an input box where scripts can be written and executed to take respective actions in the SceneGraph.
- Execute: This button allows executing the script written in the Command box.

Listener – records all the actions done in designer in command form. User can save or copy the commands from the listener window and paste into other scene(s) to perform the same set of actions.

- Record Output: When checked, enables the listener to record the activities performed in the SceneGraph.
- Save Output: This command is used to save the script in given file format.

Clear Output: This command clears the recorded output script.

Action History

The Action History window (F12) displays a list of actions that have been executed by the user. By clicking on a specific action in this window, users can move back to an earlier stage in the creation process for reference or to undo a previous action.

Alternatively, users can right-click on an action to toggle between its effect on the outcome.

The number of actions stored in the stack can be set in the Preferences dialog under File > Drone Designer.

Do any of the following to open this window:

- On the Window menu, click Action History. Or Press F12 from the keyboard.



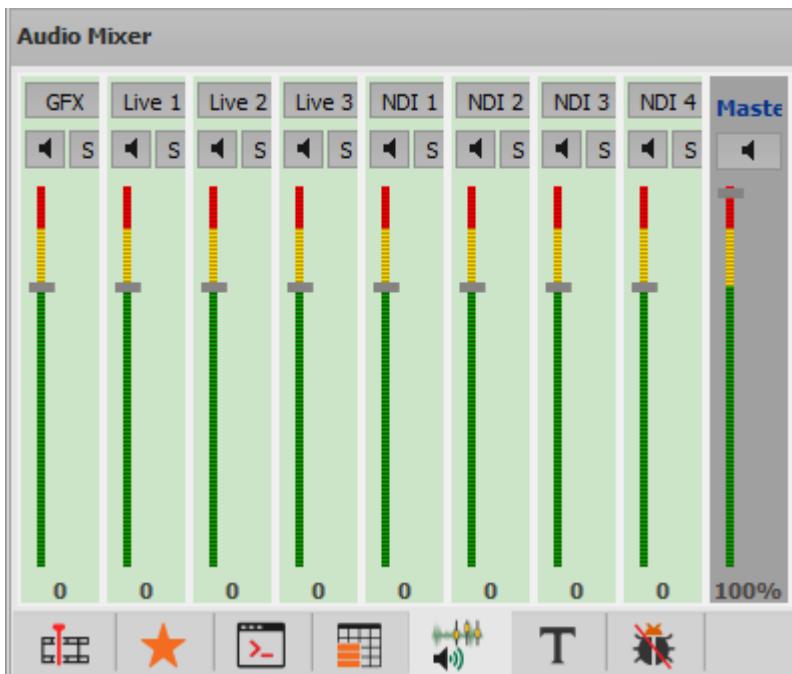
Note: Click Clear List to clear all the action history.

Action History (Shift+F3)	?	⋮
Scenegraph		
← Create Standard 1		
← Create Sphere 1		
← Create Cube 1		
← Create Text3D 1		
← Add Sphere 1 to selection		
← Sphere 1.Translation		
← Add Cube 1 to selection		
← Cube 1.Translation		
← Add Text3D 1 to selection		
← Text3D 1.Translation		
← Add Sphere 1 to selection		
← Sphere 1.Translation		
← Add Cube 1 to selection		
← Cube 1.Translation		
← Cube 1.Translation		
← Cube 1.Scale		
← Clear selection		
← Select multiple objects		
← Text3D 1.Translation		
← Text3D 1.Translation		
← Text3D 1.Translation		
← Set animation key		
← Execute command		
← Text3D 1.Translation		
← Text3D 1.Translation		
← Text3D 1.Translation		

Audio Mixer

Audio control is designed to allow the user to control the audio of Live inputs in the Designer interface. The design interface is divided into different canvases and user can assign control of these input sources in various canvases where the input source is imported.

Default Canvas is the primary canvas which will host all the inputs by default. User can assign inputs to other canvases using the Checkbox control in hierarchy when the object is added to the Viewport.



Each Canvas can control up to 8 Audio sources simultaneously. However, a combo box is available to change the input source to be controlled per audio channel. User can select any input from the available input sources added in the design scene.

Click on button in the lower dock window tabs to reveal the Audio Control.

There are 2 Sections of the Audio Control interface.

1. Canvases
2. Audio Control Area

Canvases list down section lists the number of canvases made in the scene. User can click on the canvas to reveal the control interface for that canvas. By Default, there are 8 Channels and 1 Master Control for each canvas.

Each Control Channel has the following controls.

1. Audio Level Bar - showing a Slider to controlling the audio for the respective channel.
2. A sectioned bar showing Green, Yellow, and Red colors depicts the audio level for the respective channel. Green shows controlled audio level, Yellow shows Moderate and Red depicts a high audio level per audio source.
3. A number table from (-60db) to (0db) is also displayed to numerically identify the audio level for the respective channel.

4. Click on  button to reveal Input Channel Grid where user can change the input channel map and re-assign the audio channels as required. The Input Grid shows channels in Index from 0 to 7 allowing user to map and control up to 8 channels of the input source.
5. Is Solo button. When enabled, it isolates the selected Audio Channel in the canvas and mutes all other Audio Channels. When enabled the Grey Headphone Icon turns Red. Only One Audio is isolated at one time. If more are required to be isolated from the set, use the Mute button to Unmute the respective channel.
6. Is Mute button. When Enabled, it turns Red and Mutes the respective Channel. User can Mute multiple channels as well.
7. User can click on the Mute icon in the Master Channel to Mute or Unmute All channels of the respective canvas.

Multiple canvases in the Audio Control Area are opened as tabs and user can close each tab or open more by clicking on the respective canvas in the Canvases list section.

The Master control controls the combined Audio of the respective Canvas. So, audio level all the inputs of the relative canvases can be controlled.

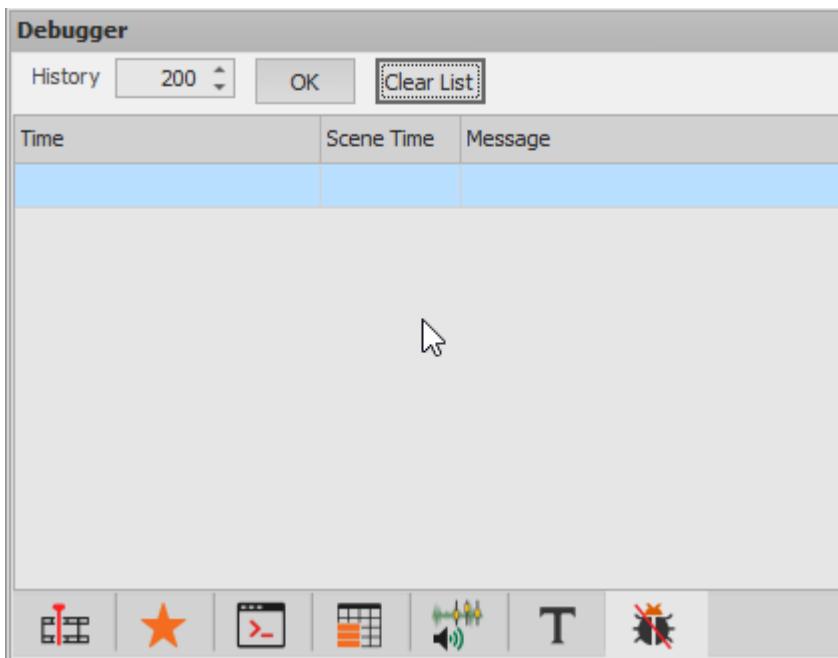
Controlling the Audio

Audio Mixer controls the audio levels of all the Live graphics inputs, NDI inputs connected to the system. Through the audio mixer, users can control the volume level and mute the input channel. Channel routing can also be done through an audio mixer.

1. Select or browse the input channel in the channel box.
2. Modify the audio input by scrubbing the audio bar up and down.
3. To mute any channel, click on button  M.
4. To play the audio of any one channel and mute all others, click on  button S.
5. To control the audio of all the channels, the user can use the MASTER channel.

Debugger

Wasp Debugger is implemented in this version. This allows user to check messages from Add-ins with require option. This is helpful in tracing the data update from Add-ins. User can use the Filter row to filter messages based on system Time, or Scene Time.



Debugger window shows the debug messages written in the scene editor code.

User can write all debug messages using the following command.

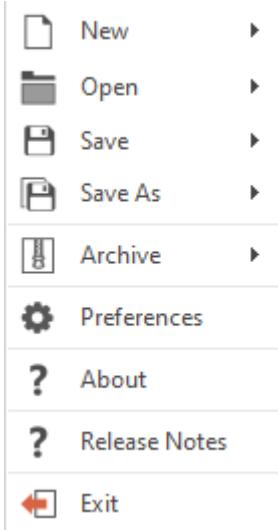
```
WDebug.WriteLine();
```

There are 3 Columns in the Debugger window.

1. **Time** – Displays the time of the message
2. Scene Time(in Frames) - displays the frame number of the scene when the trace line was written.
3. **Message** – Displays the debug message received from the code.

Menus

The  Menu bar is equipped under the Title bar of the application window. Each menu uses standard Microsoft Windows conventions and contains the relevant command buttons to perform its respective tasks. The default keyboard shortcut is mentioned next to the commands whichever have it. It helps using the application more efficiently.



New

- Blank Scene (CTRL+N): Creates a New Scene.
- Style Sheet: Opens Style Sheet Dialog box to open premade startup templates. These Style Templates are fully functional templates that you can use as a starting point for your design project.

Open

- Open Scene (CTRL+O): The 'Open [Default Location Name]' dialog box appears as you click it. Locate the scene and click OK to open it.
- Open Template (CTRL+E): It opens up the 'Waploader' (C:\inetpub\wwwroot\KernelController\Wasploder, the default location on the server where all the templates are saved) to select an existing template.
- Open Layout Template: Opens the template dialog box to open an existing template as layout scene. The layout scene will open on the topmost layer within the design viewport. It acts as a reference for adjusting and positioning the working scene based on the layout scene.
- Open Layout Scene: Opens the Open Scene dialog box to open an existing local scene as layout scene. The layout scene will open on the topmost layer within the design viewport. It acts as a reference for adjusting and positioning the working scene based on the layout scene.
- Open Style Sheet: Opens the Open Style Sheet Dialog Box to open an existing Style Sheet template. User can modify an existing Style Sheet template and Save it with changes or Save it as a New Style Sheet template for future use.

Save

- Save Scene (CTRL+S): It opens 'Save As [Default Location Name]' dialog box. Browse a location under Local Folder section, type the file name in the File Name box, and then click Save.
- Save Template (CTRL+SHIFT+E): It opens 'Save Template' dialog box. Select a folder to save the template in Wasploder root folder (This is a default sharing location.) This dialog box has Back, Up, Create Folder, Delete, Views, etc. buttons for your convenience.
- Save Style Sheet: Save the scene as a theme for local use. Save Version (CTRL+H): Version control is very helpful when working on a design, where you need to have various options for color scheme, shapes, etc. primarily to present it for approval.

Save As

- Local SG: Opens up the window with the 'Scenes' folder from the mapped X: Drive, to select an existing .wsl or any Local SG file (for the very first time you save the file) after that whenever you save the file it will keep overwriting the modifications.
- Template (CTRL+ALT+E): Saves the scene as a template in the WaspLoader root folder. You can customize the folder under WaspLoader and save the file there.
- Style Sheet(CTRL+ALT+S): Saves the scene as a theme.

Archive

- This gives us the option to archive Scenes/Templates from the Local machine/Server to a desired destination.

Preferences

- The Preferences window is where stores many of its settings. There's general display and viewport settings, device param, server settings, track view, output and more..

About

- You can see the version of Designer you're running, about the license expiry, view legal notices, learn more about the installed DLL versions report and more.

Release Notes

- You can find the new features and enhancements.

Exit

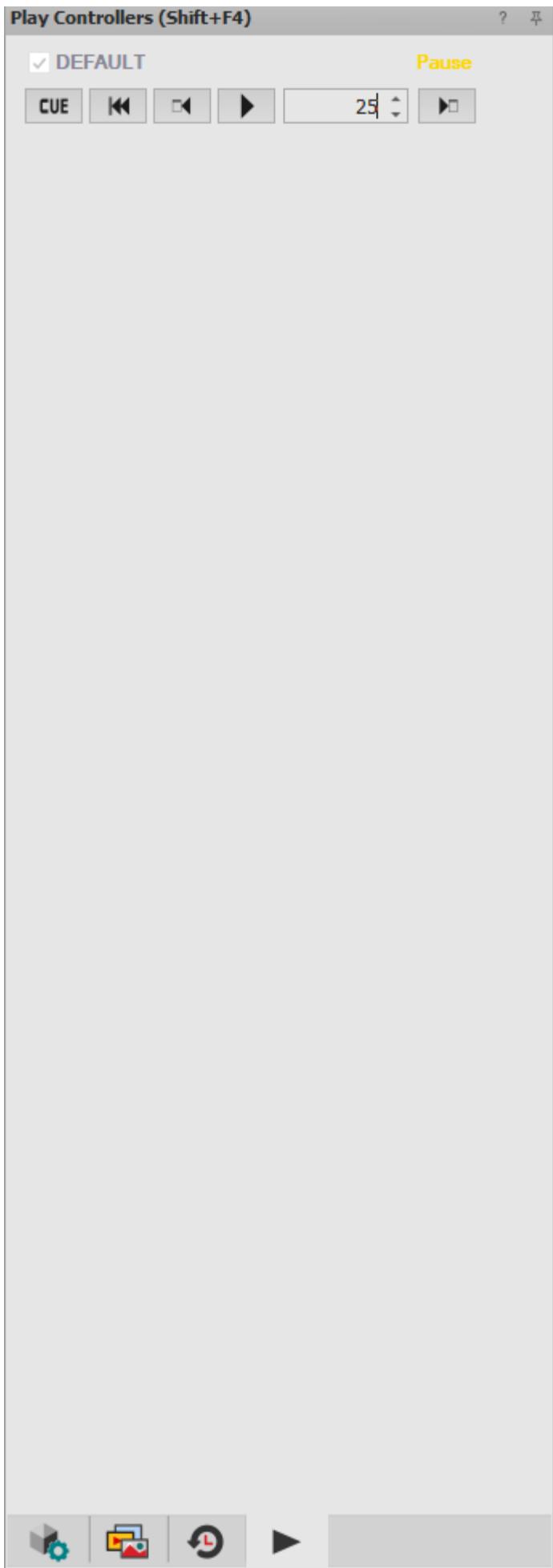
- Exit from your application.

Play Controllers

Play Controllers window displays graphic playback controls for each custom controller timeline created within a scene. Each of these controllers can be linked to Scene Controllers in the Data-Entry interface used in the Sting Client. Additional Players get added for Page Controller object when added to the scene.

There are following function controllers with reference to the shown illustration:

- Cue: Click this button to reload the scene and cue it at frame '0'.
- **Go to First Frame:** Click this button to go to the first frame.
- Go to Previous Frame: Click this button to go to the previous frame.
- Play: Click this button to play the scene.
- Pause: Click this button to pause the scene.
- Frame Text Input Box: Displays the current frame.
- **Go to Next Frame:** Click this button to go to the next frame.
- Status: Green colored text above the Go to Next Frame button shows the scene status whether it is in play mode or paused.



Render Order

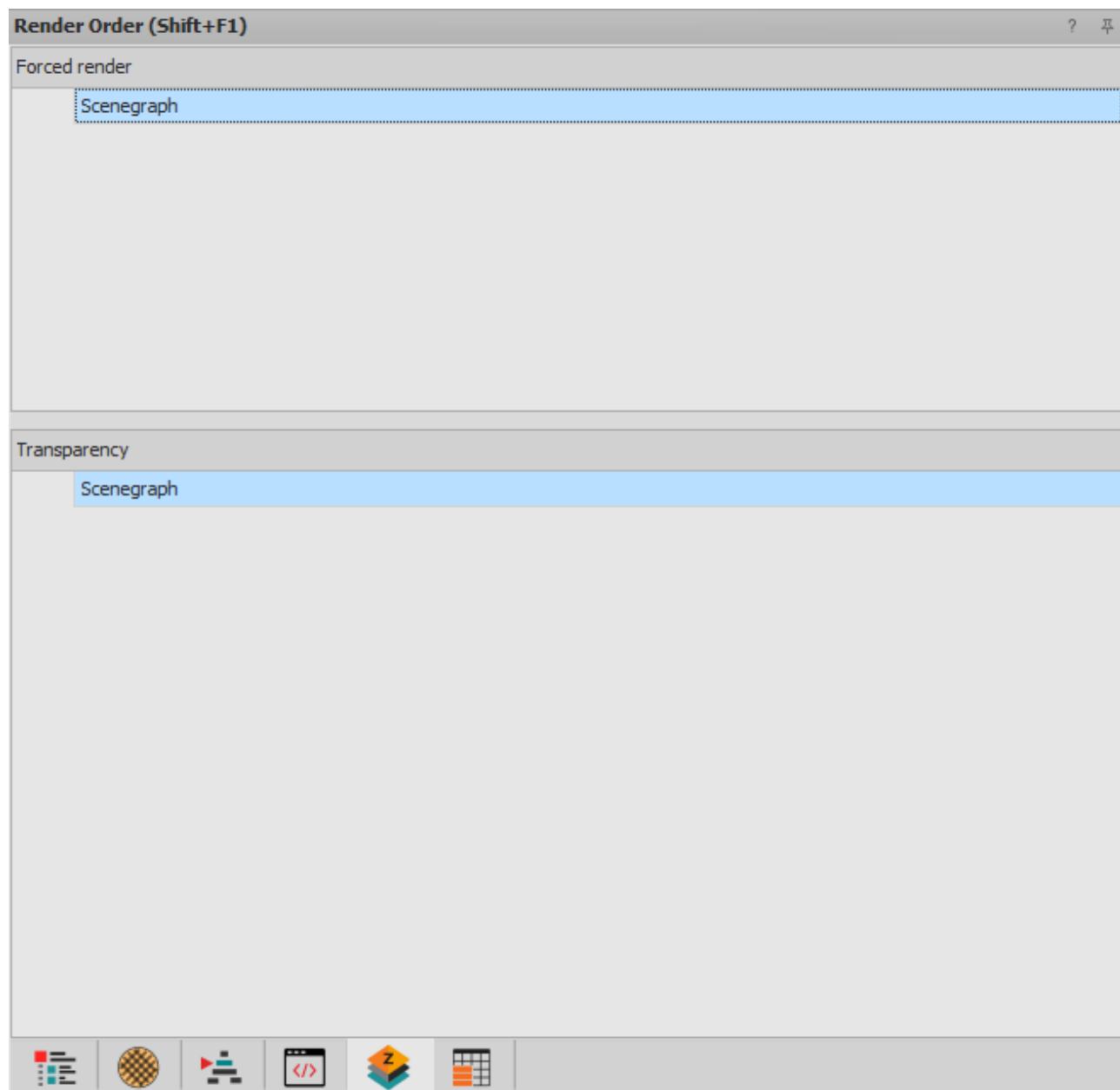
Render Order displays the objects in a Scene Graph in a specific order according to the requirements. A common problem in real-time 3D graphics is the rendering of transparent/translucent objects that share an intersection at Z-axis. The Render Order window provides two methods to overcome this problem:

The Render Order window provides two methods to overcome this common problem as follows:

1. Forced Render: Users can define a forced/imposed order of rendering for objects with various transparency values such that a "forced" object will be rendered on top of any other translucent object.
2. Transparency Order: This method takes care of priority shifting between translucent objects by automatically re-aligning the render-stack.

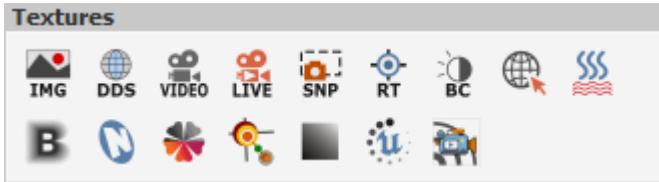
Do any of the following to open this window:

- On the Window menu, click Render Order Or Press "SHIFT+F1" from the keyboard.



Texture Pool

The Texture Pool (F6) window lists all the textures used in a scene and provides access to UVW mapping controls.



Texture - This section contains buttons for all supported texture types to import or insert and apply on the object(s) in a Scene Graph.

1. **IMG (Image):** Standard image formats like jpg, .png, etc.
2. **DDS:** A "6-sided, cubic Direct Draw texture" in DDS format stores six images forming a cube, showing various views.
3. **Video:** Standard A/V formats like .avi, .mov, .wmv, etc.
4. **LIVE (Live Video):** It can handle two live video inputs.
5. **SNP (Snapshot):** Real-Time output grabs. It is especially useful for refraction's.
6. **RT (Render Texture):** It maps 3D output to specify surface.
7. **BC (Brightness/Contrast):** It provides basic color correction controls.
8. **Browser:** Allows inserting web link.
9. **Water Texture Effect:** A texture to create Waster effect.
10. **Blur Texture:** This texture type can be used for real time implementation of Depth of Field (DOF) or for the real time implementation of Screen Space Ambient Occlusion (SSOA).
11. **Netwrok Stream Texture:** Enables Wasp3D to fetch or stream texture data from an NDI network stream. There are upto 8 input Ports available per engine. User can configure the Input Source per server using the ServeDef configuration for the respective Server using Sting Client or in Drone Designer changing the Memory Inputs in the Hardware I/O tab of Drone Designer Preferences window.
12. **Remap Texture:** Allows user to change the tint of the texture based on Shadow, Highlight, and two level of Midtone colors.
13. **LensFlare Texture:** Allows user to add Lens flare in the Wasp Scene. It uses an object to track the movement of the lens flare. User can specify colors and strength of lens flare using the parameters.
14. **Gradient:** A custom texture to create Gradient effect using four color gradients in four gradient types (Rectangle, Radial, Conical, linear).



15. UBuffer Texture: Ubuffer texture is added to support render buffer import from Unreal Engine. The camera data from the VSet Add-in is directly sent to Unreal Engine 4. Hence Unreal Engine 4 camera can be controlled by using VSet Camera Controls.



16. Placeholder Texture: Placeholder Texture enables a user to select any input to be displayed in the object it is applied as a texture. Live, NDI, Image, and Video can be selected to be applied to this texture. Users can use the Placeholder control in the Client application to modify the input right from the client interface without the need to wire or apply any variables. 8 distinct placeholders can be applied in an engine at one time. As the placeholder selection is at the engine level, any change in one placeholder would also apply to the other placeholder which is using the same placeholder channel.

Standard Attributes

Standard

Opacity	251	Mip Level	0.00	
U	V	W		
Offset	0.000	0	0.000	
Scaling	P	1.000	0	1.000
Rotation	0.000	0	0.000	
Center	0.500	0	0.500	
Reset All				

This area contains the texture manipulation parameters - UVW Translate, UVW Rotate, UVW Scale and UVW center Rotation.

Parametric Attributes

Parametric

Default

File Name	D:\Graphic_Template_Icon_-
Width	32.000
Height	32.000
Play Control	0.0000,-1.0000,0.0000,6630...
Totalframes	1
Equirectangular	<input type="checkbox"/>
Tag	
Rows	1
Columns	1
Depth	0
Landscape	<input type="checkbox"/>
Mip Map	<input type="checkbox"/>

File Name: Shows the file path of the texture file.

Width: Displays the width of the texture in pixels.

Height: Displays the height of the texture in pixels.

Play Control: Provides an interface to control the video texture. This includes options such as seek, play (by specifying the number of times), infinite play, and setting the movie length.

Total Frames: Displays the total number of frames in animated images like GIFs.

Equirectangular: Used for cubic projection or spherical panorama viewers, supporting 360° view textures.

Tag: Serves as a variable placeholder to hold values for the object. Users can specify object names in the "Tag" and wire properties of other objects by specifying "Object.Tag" as the target.

Rows: Likely refers to the number of rows used in a grid for textures, possibly for slicing or tiling the texture into multiple parts.

Columns: Likely refers to the number of columns used in a grid for slicing or tiling the texture.

Depth: Refers to the depth attribute of the texture, possibly for 3D textures or layers.

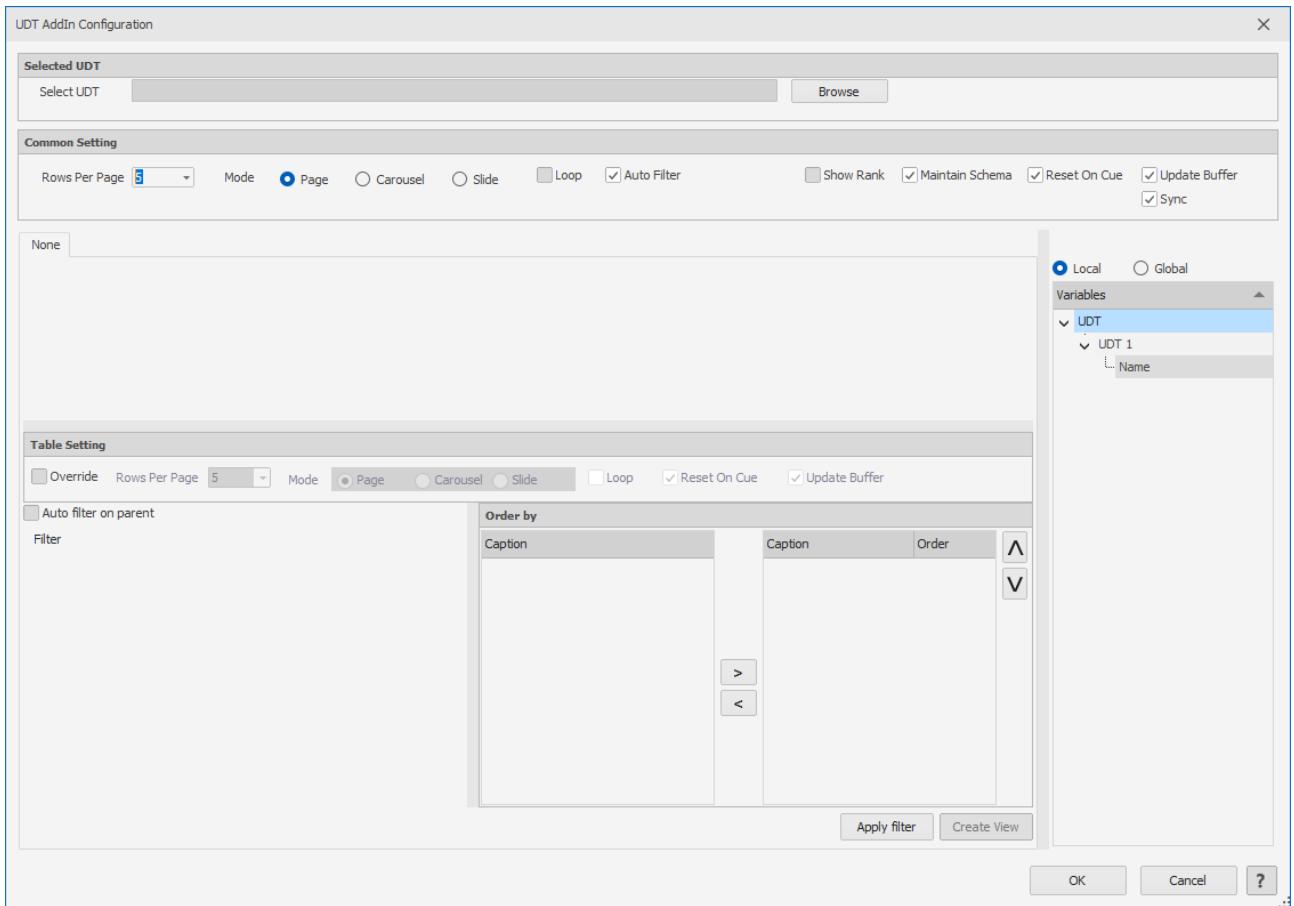
Landscape: When checked, it indicates the texture orientation is in landscape mode.

Mip Map: Refers to the process of using mipmaps, which are pre-calculated, optimized sequences of images for textures, commonly used to improve rendering performance in 3D graphics. It might indicate whether mipmaps are enabled for the texture.

Provides texture specific parameters, which changes the parameters of the respective texture. For eg. Video Texture has Video Controls to control the Play, Pause & Pause/Stop of the video. Brightness/Contrast Texture has Brightness, Contrast, Hue and Saturation parameters to control the texture.

UDT Pool

An acronym for User Defined Tables, designed primarily to facilitate functionality like that of a database. It provides users access to a collection of tables that can be customized to create hierarchical, category-based data entry and storage interfaces.



Browse: search for the file and add from any location on local system or network systems.

Common Settings: These settings are applied on all the table sheets in the selected file. To override the common settings user can specify settings per table using the table settings.

- Row per page: select number of rows of data to be extracted for single display of data. The Row per page is set according to the number of items to be displayed in one data set of graphic. When user wants to display a team sheet of soccer for 11 players, the row per page would be set to 11. and when showing data for a Map where all constituencies are required to be colored, the All option is selected for it.
- Mode: User can select the relevant mode to get the updated data when the Record Move action is called. The page iteration is divided into sets of items as per the number defined in the Rows per page value.
 - Page: For e.g. There are 20 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6 to 10, 11 to 15, 16 to 20.
 - Carousel: This mode is designed to move the data iteration for carousel graphic movements. The Carousel mode best works with the Menu group control. User can create many styles of Menu Groups for touch or standard graphics page data functionality. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 6,2,3,4,5 then 6,7,3,4,5. the change is like this because the 1st item will exit and will become the 6th item, similarly, the send item will exit and become the 7th item. so the last page which would be the 6th page would show, 6,7,8,9,10. and the sequence would continue as per the menu group functionality.

- Slide: This mode is designed as another menu mode to accommodate data flow from the set in a slide show form. This mode is much useful when the object animation is not required, but data should change. For e.g. There are 10 rows of data in a file. and Row per page is 5. The 1st page will show rows 1 to 5 next page will show Rows 2,3,4,5,6 then 3,4,5,6,7 and last page set would be 6,7,8,9,10.
- Loop: Repeats the data once reached end of the records. The end of records is calculated based on the selected mode.
- Auto Filter: When enabled, filters the UDT as per the Visible Rows.
- Auto Update: When enabled, it updates the data of the currently displayed page on-air, else the data updates are displayed when the data page is called next time or in loop. The updates if done in the rows in 2nd page are displayed when the record move action is called.
- Show Rank: Adds a Rank column in the existing data set. This is useful to see the row numbers and to filter the data based on row number if there no similar column in the data.
- Maintain Schema: Locks the data assignment with the column. This is helpful in case rows are added in between the existing data set after the data is mapped to the graphic.
- Reset on Cue: When enabled, sets the data set to page 1 when the scene is cued.
- Update Buffer: When enabled, the data updates are fetched in page and updated on-air. When unchecked, the update is fetched but the text is not updated on-air. The data is updated on-air, when the update buffer action is called using an action set or at required frame on the timeline. The case is useful when the data is wired, but only requires to be updated on demand. User can use this option to achieve data update on demand or in sync with animation instead of the Set Param command.
- Sync: When enabled, the data updated in the table and updated as soon as the user enters the data in the UDT.

Data Window: Displays tabs based on sheets available in the UDT. Other sheets are only loaded if there is some data in it. Data is displayed in form of columns.

Table Setting: The parameters in Table setting are similar to what is available in the common settings. They are used to override the common settings for the respective table data. Please see the parameters defined in the Common parameters for more info.

Variables: Displays list of variables in the scene which can be dragged to the filter column to filter the data as per the data point in variable.

- Local: Gives the list of local variables which are only available in the scene in which it is created
- Global: Gives the list of global variables which are available/accessible in any scene

Filter: Filters the data table as per the defined filter. Click on the + button to add filter.

To filter table data with static or variable value adding conditions like “begins with”, “ends with”, “equals to”

- Order by: Re-arranges the content of the sheets in the order specified.

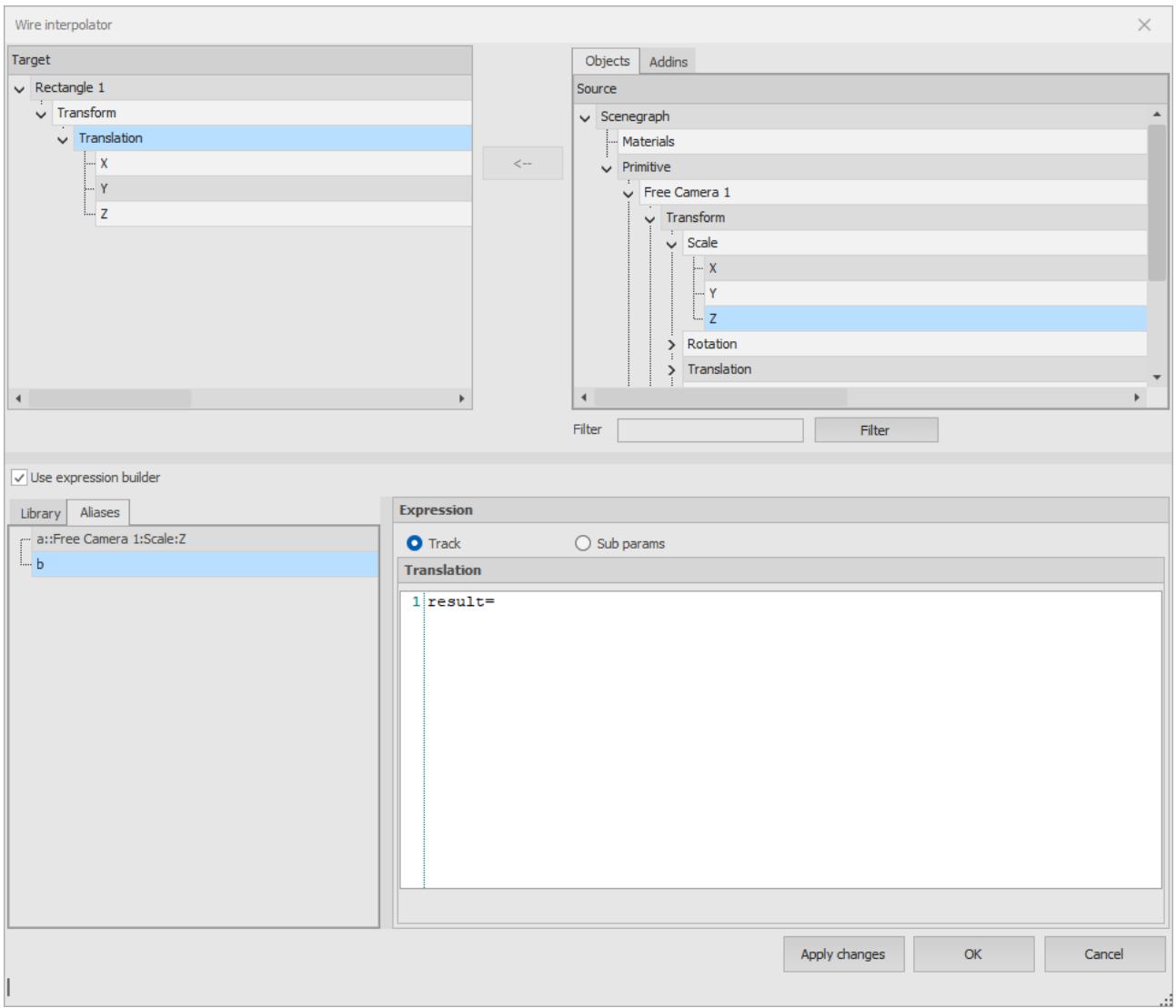
Apply Filter: Applies the selected settings to the Data Table(s).

Create View: Creates a clone of the selected table. User sometimes require to call a data file more than once to specify different filter parameters. This helps user to create custom views of the same table and prevents the user to create multiple Add-ins for the same data source.

Wire

Wiring means connecting or linking. Wasp3D Supports wiring of various Object, Texture or Material Parameters, Variables, Data from Tables, Animation Tracks, Transformations etc.

User can create a direct connection between any property/variable by drag and drop of variable to the respective parameter, selecting the Wire Interpolator or enter an expression or script in the Expression Wiring control shown below to do script or expression based wiring. Wasp3D supports JavaScript based wiring expression.



Wire Interpolator

The Wire Interpolator is a tool for connecting parameters and transformation properties of objects in a scene, facilitating creative and data management processes. Users can establish connections between parameters, enabling one parameter to influence another, resulting in interactive scene elements. For instance, you can link the Length of a text to the visibility of a Rectangle; when the text length is zero, the Rectangle's visibility becomes zero.

The Wire Interpolator simplifies operations by presenting a hierarchical display of scene objects and textures, categorizing their parameters and transformation properties for easy reference. Users enjoy flexibility in choosing and connecting specific parameters to a target parameter, granting control over interactions between various scene elements.

Target

"Target" is an item, like an object, parameter, or variable, within the scene that you want to connect or link with another object or parameter. This Target acts as the endpoint where data or influence, usually chosen by the user as the source parameter, is directed. When a user right-clicks on a parameter or

variable to create a connection, it's identified and displayed as the Target, indicating that it's the destination for the data or influence from the source.

Objects

Displays the list of objects and textures in the scene. All the parameters and Transformation properties of each object are categorized and displayed in a hierarchical structure. The user can select the parameter to be wired with the target parameter.

Addins

Displays the list of Addins and Variables in the Scene that can be connected to the Target item.

Filter

Isolates the parameters or values specified in the Filter Text Box. User enter a term and press on Filter button to filter the results in the objects and Addins list.

Use expression builder

Toggles the Express Builder functionality. When Enabled, allows user to write script or formulas based on multiple alias as dragged from the list of Objects and Addins. Multiple Variables, Parameters can be used in the Expression builder to provide a calculated value in the target.

Library

Displays the list of premade scripts that can be used as expressions. These scripts are prepared by Wasp3D and comments are provided in the script for easy deployment. When a script is clicked, it is displayed in the Expression. Specify the aliases required in the script and get the final result.

Aliases

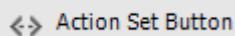
Alias refer to a user-defined alternative name or label given to a parameter, object, or variable in the scene. It provides a more user-friendly or easily recognizable reference for that element within the Wire Interpolator, making it simpler to work with and manage. A Parameters or a Variable which is to be used in the expression requires to be added as an Alias. The Alias is created in alphabetical order starting from a. User can drag multiple Parameters or variables in Alias and use them in the Expressions.

Expression

This script area allows users to define how parameters are wired or interconnected within the software. The Wire Interpolator supports all JavaScript functions, giving users the flexibility to create complex expressions and logic for controlling and manipulating the parameters.

Users can apply the Expression as a whole to establish a relationship between source and target parameters. Additionally, they have the option to apply the Expression to individual sub-parameters within the target, allowing for fine-grained control and customization of the wiring process.

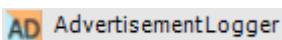
Action Set Button



Add-In Page Player



Advertisement Logger

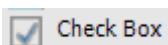


This wasp control will get the specific text values for AdvertisementID and AdvertisementSlug name for the template. The property Modifier will help user to define scope of execution ability to Public, Protected, Protected Internal and Private.

The list of the advertisement concerned is to be stored in collection wasp control.

You can observe the object of the advertisement control on the bottom end of the active Form.

Check Box



The Wasp3D Forms CheckBox control indicates whether a particular condition is on or off. It is commonly used to present a Yes/No or True/False selection to the user. You can use check box controls in groups to display multiple choices from which the user can select one or more.

The check box control is similar to the radio button control in that each is used to indicate a selection that is made by the user. They differ in that only one radio button in a group can be selected at a time. With the check box control, however, any number of check boxes may be selected. A check box may be connected to elements in a database using simple data binding. Multiple check boxes may be grouped using the GroupBox control. This is useful for visual appearance and also for user interface design, since grouped controls can be moved around together on the form designer.

The CheckBox control has two important properties, Checked and CheckState. The Checked property returns either true or false. The CheckState property returns either CheckState.Checked or CheckState.Unchecked; or, if the ThreeState property is set to true, CheckState may also return CheckState.Indeterminate.

In the indeterminate state, the box is displayed with a dimmed appearance to indicate the option is unavailable.

Clock

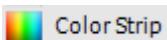


This control in Form work in designer mode this clock control gets user the current system time and converts in string according to the selected time format and culture language. You can associate this control to the user tag or custom tag as required by the Template designer in the wsp file.

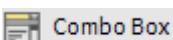


Note: You can't change the data of this control, as by default the data is set to the system time.

Color Strip



Combo Box

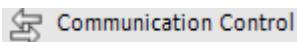


A Combo Box displays an editing field combined with a list in the box allowing the user to select from the list or to enter new text. The default behavior of Combo Box displays an edit field with a hidden drop-down list. The drop down style property determines the style of combo box to display. You can enter a value that allows for a simple drop-down, where the list always displays, a drop-down list box, where the text portion is not editable and you must select an arrow to view the drop-down list box, or the default drop-down list box, where the text portion is editable and the user must press the arrow key to view the list.



Note: The Combo Box also takes items from the database through Data Source property and the corresponding properties like Display Member and Data Bindings. See more examples further in this write-up.

Communication Control



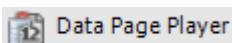
The Communication Addin is a tool used to facilitate the exchange of data using the TCP/IP ports via various communication links. It allows users to select a communication link, send single or multiple commands, and transmit predefined variables such as keys and values. This addin allows user to send the communication across all the Wasp3D Modules.

- Send Key: This command is used to send data over a selected communication link. Once a command is received, the defined set of actions are carried forward in the receiver scene.
- Command: Allows users to send a command via the communication link. The command is received and triggers the corresponding action. The data is displayed in form of a Data Table.
- Select Link: Enables users to select the desired communication link for transmitting data.
- Send: Option allows users to send a single command through the communication link for communication purposes.
- Send Multiple: Option allows users to send multiple commands simultaneously over the communication link.
- Key, Value 1, Value 2, Value 3, Value 4, Value 5, Value 6: These are predefined variable types that are used for sending commands through the communication link. When receiving multiple values, the user is required to use Match Variable Command and take appropriate actions accordingly. The Values are received in the Command Data Table. The actions can be called on Value Change, Row Value Change or on Data Update events of the Command Data Table.

Countdown Timer

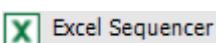


Data Page Player



This control facilitates the user in running pages defined in the PageControl. Drag and drop this control in Design Controller. While we run this template in Sting on Air Encapsulator we can see the buttons of each page on the control, which will help the user in selecting the page manually to be transmitted at the On-Air PlayOut level.

Excel Sequencer



File Dialog

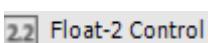


This control will can be created directly on the form itself. Inheriting from this class is also possible. When we use this control, a dialog box appears on the user form to select a file. `FileDialog` is a modal `openFileDialog` box; therefore, when shown, it blocks the rest of the application until the user has chosen a file. When a dialog box is displayed modally, no input (keyboard or mouse click) can occur except to objects on the dialog box. The program must hide or close the dialog box (usually in response to some user action) before input to the calling program can occur.

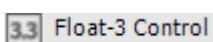
Float Control



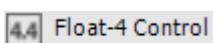
Float-2 Control



Float-3 Control



Float-4 Control



Flow Layout Panel



Graph Control



Group Box



This control displays a frame around a group of controls with or without a caption. Use a Group Box to logically group a collection of controls on a form. The group box is a container control that can be used to define groups of controls.

The typical use for a group box is to contain a logical group of Radio Button controls. If you have two group boxes, each containing several radio buttons, each group of buttons is mutually exclusive, setting one option value per group.

Controls to the Group Box can be added by using the Add method of the Controls property.



Note: Only controls contained within the group Box control can be selected or receive focus.
The entire group Box itself cannot be selected or receive focus.

Image Editor



Label

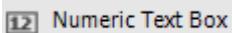


Label controls are typically used to provide descriptive text for a control. For example, we can use a Label to add descriptive text for a Text Box control to inform the user about the type of data expected in the control. Label controls can also be used to add descriptive text to a Form to provide the user with helpful information. For example, we can add a Label to the top of a Form that provides instructions to the user on how to input data in the controls on the form. Label controls can be also used to display run time information on the status of an application. For example, we can add a Label control to a form to display the status of each file as a list of files is processed.

In addition to displaying text, the Label control can also display an image using the Image property, or a combination of the Image Index and Image List properties.

 **Note:** A Label can be made transparent by setting its Back Color property to Color.Transparent. When you use a transparent label, use only the current device coordinate system to draw on the container, or the Label background might paint improperly.

Numeric Text Box

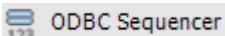


A NumericTextBox control contains a single numeric value that can be incremented or decremented by clicking the up or down buttons of the control.

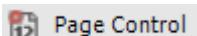
To specify the allowable range of values for the control, set the MinVal and MaxVal properties. Set the value to specify the value to be incremented or decremented to the Step property when the user clicks the up or down arrow buttons.

When the Up or Down buttons are called, either in code or by the click of the up or down buttons, the new value is validated and the control is updated with the new value in the appropriate format. The value is then verified to be between the Minimum and Maximum values and the corresponding method is called.

ODBC Sequencer

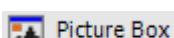


Page Control



Helps in implementing 'pages' at the On-Air PlayOut level. Based on the amount of data input, the data entry clients can create pages. This functionality helps a graphic designer to save time by creating/modifying the graphic scenes according to the content. Wasp automatically takes care of this and manipulates the scenes based on content. All the controls on PageControl work according to the data associate within them to manage multiple profiles of the same type. For instance if we need to flash the election result on screen, we can use this control to show profile of VIP candidates and their present status.

Picture Box



The PictureBox is a control used to display a picture on the data entry form. The picture to be displayed is determined by the Image property, which can be set at run time or at design time. TheSizeMode property controls how the image and control fit with each other. For more information, see if you use the Wasp3D Forms PictureBox control on a form, you can set theSizeMode property to it through property sheet:

Align the picture's upper left corner with the control's upper left corner

Center the picture within the control

Adjust the size of the control to fit the picture it displays

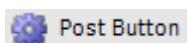
Stretch any picture it displays to fit the control

Stretching a picture (especially one in bitmap format) can produce a loss in image quality. Metafiles, which are lists of graphics instructions for drawing images at run time, are better suited for stretching than bitmaps are. A specially added feature of this control is AutoOpenFileDialog on the double click event of the PictureBox, we one can select the picture from the source easily. This will improve the working speed. Apart from this, Wasp3D PictureBox has another function to get pictures i.e. Image Bypath. This function will help you to get the picture directly from the source path.

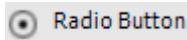
To set theSizeMode property at run time Set PictureBox.SizeMode to Normal (the default), AutoSize, CenterImage, or StretchImage. Normal means that the image is placed in the control's upper-left corner; if the image is larger than the control, its lower and right edges are clipped. CenterImage means that the image is centered within the control; if the image is larger than the control, the outside edges of picture are clipped. AutoSize means that the size of the control is adjusted to the size of the image.

StretchImage is the reverse, and means that the size of the image is adjusted to the size of the control.

Post Button



Radio Button



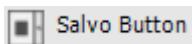
The Radio Button control can display text, an Image, or both. When the user selects one radio button within a group, the others clear automatically. All RadioButton controls in a given container, such as a Form, constitute a group. To create multiple groups on one form, place each group in its own container, such as a GroupBox or Panel control.

Radio Button and CheckBox controls have a similar function: they offer choices to a user. The difference is that multiple CheckBox controls can be selected at the same time, but radio buttons are mutually exclusive. Use the Checked property to get or set the state of a RadioButton. The radio button's appearance can be altered to appear as a toggle-style button or as a standard radio button by setting the Appearance property.

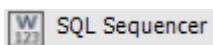
RTF Text Box



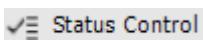
Salvo Button



SQL Sequencer



Status Control



Tab Control

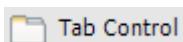
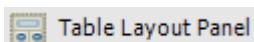
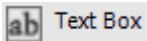


Table Layout Panel



Text Box



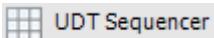
The Text Box control allows the user to enter text in an application. This control has additional functionality that is not found in the standard Windows text box control, including multiline editing and password character masking. Typically, a Text Box control is used to display, or accept as input, a single line of text. You can use the Multiline and ScrollBars properties to enable multiple lines of text to be displayed or entered.

Set the AcceptsTab and AcceptsReturn properties to true to allow greater text manipulation in a multiline Text Box control. You can limit the amount of text entered into a TextBox control by setting the MaxLength property to a specific number of characters. Text Box controls can also be used to accept passwords and other sensitive information. You can use the PasswordChar property to mask characters entered in a single-line version of the control. To restrict text from being entered in a TextBox control, you can create an event handler for the KeyDown event in order to validate each character entered in the control. You can also restrict all entry of data in a TextBox control by setting the ReadOnly property to true.

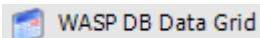
UDT Data Editor



UDT Sequencer



Wasp DB Data Grid



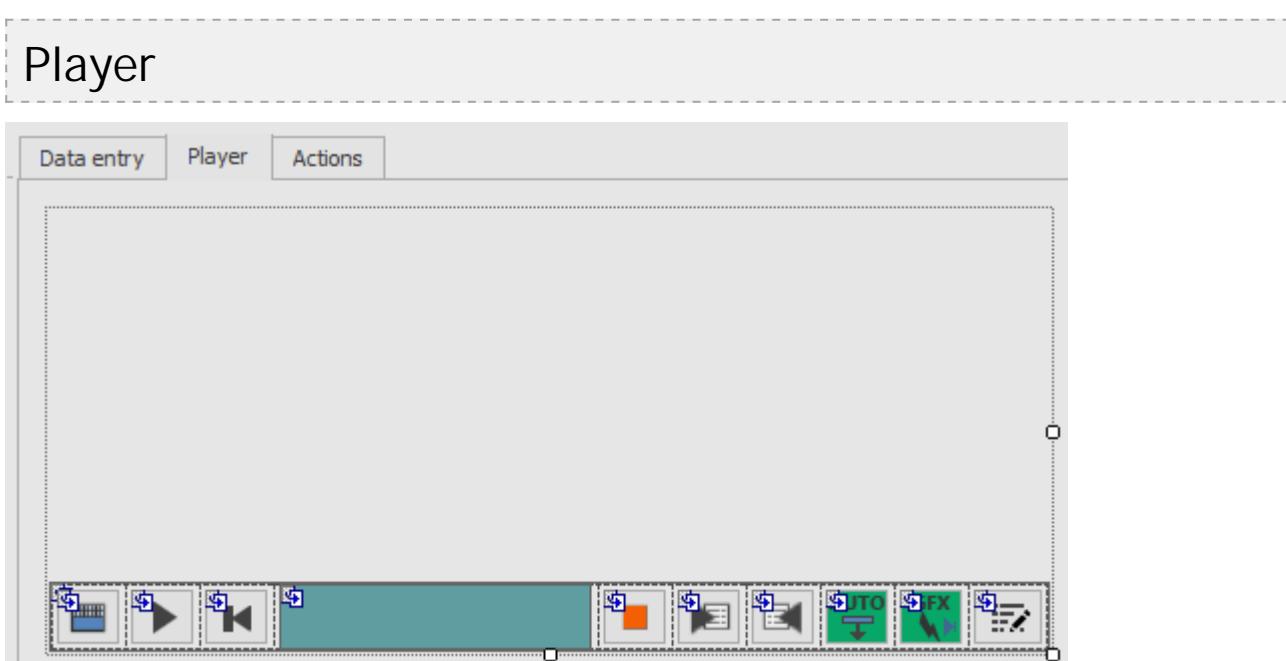
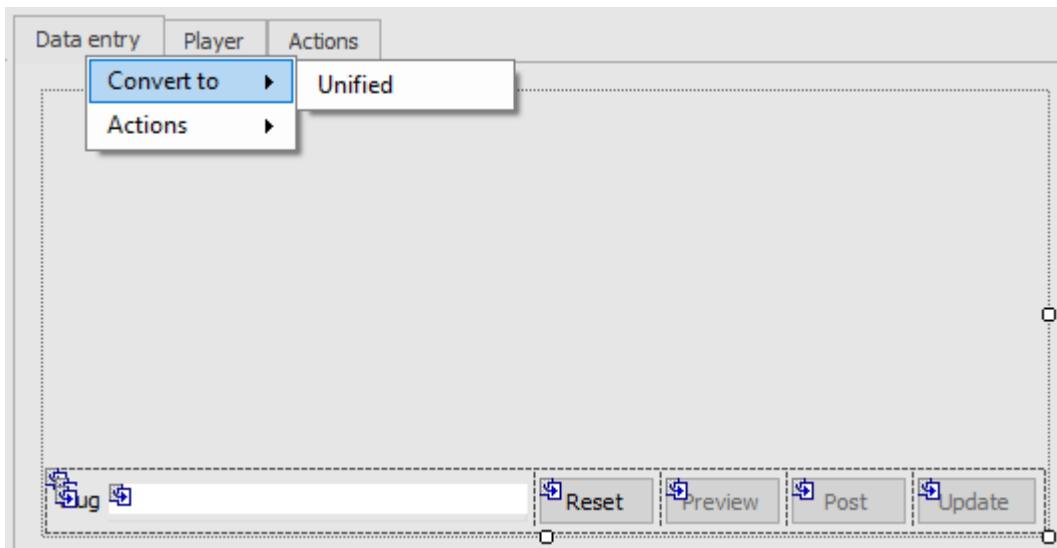
This control consists of the DataGrid pages, which can be associated with the data source with the WaspConnection object. The WaspDbDataGrid provides a facility of linking the user tags of wasp nodes to the corresponding data fields of the database. The WaspDbDataGrid displays links to tables on each page. You can click on a page link to navigate to the particular table.

Data Entry

The Data Entry tab contains a blank form and facilitates the creation of Data Entry Forms for your templates. It allows you to add controls relevant to the template and position them appropriately.

There are two types of forms available for design. Right-click on the Data Entry Tab to access the Convert To menu to select the form type for your template.

1. **Unified:** A single form is available for both data entry and Play Controllers.
2. Standard: Two different forms are available for data entry and design player.



1. Press the Cue (F9) button to cue the template in the Render Engine.
2. When the template is cued, it would enable the Play & Stop buttons, show scene cued in the status area and disable to Cue button, as shown in the image below.
3. Press the Play (CTRL+Space bar) button to play the scene in the render engine. Once played, The Play Button changes to Pause and the status is changed to Playing and the Row color of the instance changes to Green. User can press Pause Button (CTRL+Space bar) again to pause the scene.



Note: when the scene is on Pause, the objects, textures and material which are set to animate on loop (inheriting the scene animation) do not pause.

4. If a scene is on a timed pause, it would release automatically after the set time. An Infinite Pause might require the graphic operator to play the scene, or it can also trigger as per a scene event or another timeline in the same scene e.g., a Page Controller.
5. There can be any number of Play or Pause states in a scene. The Graphic operator should consult the Graphic Designer for the scene operation and functionality.
6. User can press the Stop Button(F11) to stop the scene as desired. In a typical scenario, the scene would automatically pause and unload from the player while playing its out animation. If the scene is played out and not stopped automatically, the user can press the Stop Button and/or press the Close button at the top-right of the Player to unload the scene from the player.



Info: The functionality to Stop and Unload is controlled by a Play Complete action in the scene. It is the responsibility of the Graphic Designer to apply the Play Complete key in the design as per its animation sequence or functionality.

7. The Load Next Instance (F12) and Load Previous Instance (Alt + F12) buttons load the Next and the Previous instance in the available player. The Next button would not work on the Last instance and The Previous button would not work on the First instance in the playlist. If a player is not available the buttons.
8. **Auto Unload** button is a toggle button. When enabled it shows in Green Color which unloads the player form from the player when the scene triggers the Play Complete action and vice-versa.
9. **Trigger Next** Graphic button is also a toggle button. When enabled, it would enable the loaded template to load and play the next graphic based on the Trigger Next Graphic key applied in the template. If the Trigger Next Graphic key is applied in a template when triggered,
 1. If the player is empty - it would load the next instance in the playlist.
 2. If the player is already loaded – The template would be played.
10. F2 or the Edit Instance buttons allows the user to edit the on-air instance. Any updates done in the form gets updated on-air and also in the instance.

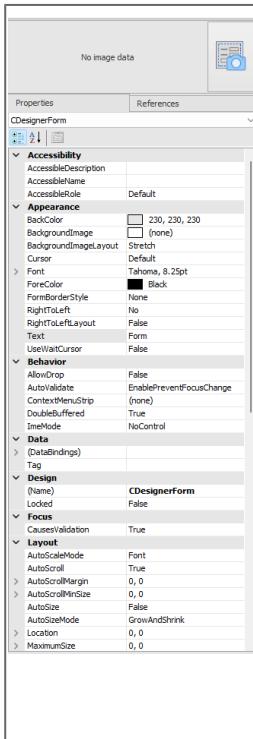
Action



Properties

The Property Sheet lists the attributes of the selected control object (class). Users can view the list alphabetically, or categorize it by attribute and map the class attributes values to their requirements. In the Events section you can call the events in the code as per code requirement

For example: You will call the Click Event in your code by clicking here



- Accessibility

- **Accessible description:** The description that will be reported to accessibility clients.
- **Accessible Name:** The name that will be reported to accessibility clients.
- **Accessible Role:** The role that will be reported to accessibility clients.

- Appearance

- **Back Color:** The background color of the component.
- **Background Image:** The background image used for the control.
- **Background Image Layout:** The background image layout used for the component.
- **Cursor:** The cursor that appears when the pointer moves over the control.
- **Font:** Font used to display text in the control.
- **ForeColor:** The foreground color of this component, which is used to display text.
- **FormBorderStyle:** Indicates the appearance and behavior of the border and title bar of the form.

> MinimumSize	0, 0
> Padding	0, 0, 0, 0
> Size	500, 220
StartPosition	WindowsDefaultLocation
WindowState	Normal
✓ Misc	
AcceptButton	(none)
CancelButton	(none)
InstanceReID	
KeyPreview	True
ShowDataPanel	False
TemplateID	
TemplateReID	
ToolTip on toolTipPlayer	
UDxnl	
✓ Wasp3D	
DefaultSlug	
Show Metadata Image	False
ValidatePost	True
✓ Window Style	
ControlBox	True
HelpButton	False
Icon	 (Icon)
IsMdiContainer	False
MainMenuStrip	(none)
MaximizeBox	True
MinimizeBox	True
Opacity	100%
ShowIcon	True
ShowInTaskbar	False
SizeGripStyle	Auto
TopMost	False
TransparencyKey	

- **RightToLeft:** Indicates whether the component should draw right-to-left for RTL languages.

- RightToLeftLayout: Indicates whether the control layout is right-to-left when the RightToLeft property is set to Yes.

- Text: The text associated with the control.

- UseWaitCursor: When this property is true, the Cursor property of the control and its child controls is set to WaitCursor.

- Behavior

- Allow Drop: Indicates whether the control can accept data that the user drags onto it.

- AutoValidate: Indicates whether controls in the container are automatically validated when the focus changes.

- Context Menu Strip: The shortcut menu to display when the user right-clicks the control.

- DoubleBuffered: Indicates whether the control will be double buffered.

- ImeMode: Determines the IME (Input Method Editor) status of the object when selected.

- Data

- **Data Bindings:** The data bindings for the control.

- **Tag:** User-defined data associated with the object.

- Design

- Name: Indicates the name used in code to identify the object.

- Locked: The Locked property determines if we can move or resize the control.

- Focus

- Causes Validation: Indicates whether this component raises validation events,

- Layout

- AutoScaleMode: Determines how this form or control will scale when screen resolution or fonts change.

- **Auto Scroll:** Indicates whether scroll bars automatically appear when the control contents are larger than its visible area.

- **Auto Scroll Margin:** The margin around controls during auto scroll.

- Auto Scroll Min Size: The minimum logical size of the auto scroll region.

- **Auto Size:** Specifies whether a control will automatically size itself to fit its contents.

- **Auto Size Mode:** Specifies the mode by which the user interface element automatically resizes itself.

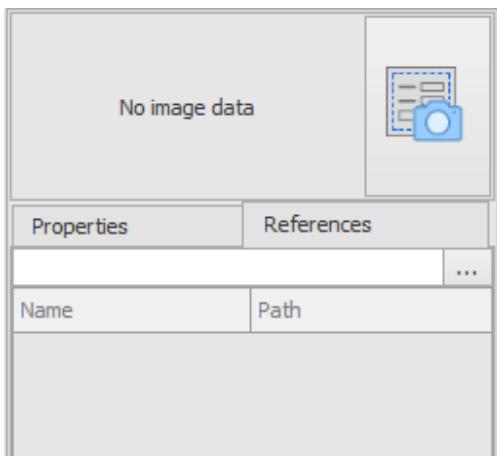
- Location: The coordinates of the upper-left corner of the control relative to the upper-left corner of its container.

- Maximum Size: Specifies the maximum size of the control.

- **Minimum Size:** Specifies the minimum size of the control.

- Padding: Specifies the interior spacing of a control.
 - Size: The size Of the control in pixels.
 - StartPosition: Determines Mne position of a form when it first appears.
 - WindowState: Determines the initial visual state of the form.
- Misc
 - AcceptButton: The accept button of the form. If this is set, the button is 'dicked' whenever the user presses the 'EBITER' key.
 - CancelButton: The cancel button of the form. If this property is set, frie button is 'dicked' whenever the user presses the 'ESC' key.
 - InstanceRefID:
 - Keypreview: Determines whether keyboard events for controls on the form are registered with the form.
 - ShowDataPanel:
 - TemplateID:
 - TemplateRefID:
 - ToolTip on tootTipPlayer: Determines the Toomp shown when the mouse hovers over the control.
 - UIXml:
- Wasp3D
 - DefaultSlug: For associating usertag of scenegraph with the control
 - Show Metadata Image: Show Metadata Image
 - **ValidatePost:**
- Window Style
 - ControlBox: Determines whether a form has a Control/System menu box.
 - HelpButton: Determines whether a form has a Help button on the caption bar.
 - Icon: Indicates the icon for a form. This icon is displayed in the form's system menu box and when the form is minimized.
 - IsMdiContainer: Determines whether the form is an MDI container.
 - MainMenuStrip: Specifies the primary MenuStrip for the Form. This property is used for keyboard activation and automatic merging in MDI.
 - MaximizeBox: Determines whether a form has a maximize box in the upper-right corner ofits caption bar.
 - MinimizeBox: Determines whether a form has a minimize box in the upper-right corner of its capton bar.
 - Opacity: The opacity percentage of the control.
 - ShowIcon: Indicates whether an icon is displayed in the title bar ofthe form.
 - ShowInTaskbar: Determines whether the form appears in Yne Windows Taskbar.

- SizeGripStyle: Determines when the SizeGrip will be displayed for the form.
- Topmost: Indicates whether the form always appears above all other forms that do not have this property set to true.
- TransparencyKey: A color which will appear transparent when painted on the form.



References

Browse:

Name:

Path:

evtEProviderAdded Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtProviderAdded m_evtEProviderAdded`

VB: `Public Event m_evtEProviderAdded As dlgtProviderAdded`

Value

Type: dlgtProviderAdded

evtEWActiveParam Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtActiveParam m_evtEWActiveParam`

VB: `Public Event m_evtEWActiveParam As dlgtActiveParam`

Value

Type: GFXWrpSceneGraph.dlgtActiveParam

evtEWAdLog Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtAdvertisementLog m_evtEWAdLog`

VB: `Public Event m_evtEWAdLog As dlgtAdvertisementLog`

Value

Type: GFXWrpSceneGraph.dlgtAdvertisementLog

CLayoutEngine.m_evtEWEEngineLoaded Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtEngineLoaded m_evtEWEEngineLoaded`

VB: `Public Event m_evtEWEEngineLoaded As dlgtEngineLoaded`

Value

Type: GFXWrpSceneGraph.dlgtEngineLoaded

m_evtEPlaceHolderProRemoved Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtPlaceHolderProviderRemoved m_evtEPlaceHolderProRemoved`

VB: `Public Event m_evtEPlaceHolderProRemoved As dlgtPlaceHolderProviderRemoved`

Value

Type: dlgtPlaceHolderProviderRemoved

m_evtEWCurrentTool Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtCurrentTool m_evtEWCurrentTool`

VB: `Public Event m_evtEWCurrentTool As dlgtCurrentTool`

Value

Type: GFXWrpSceneGraph.dlgtCurrentTool

CLayoutEngine.m_evtEWDelUnusedSg Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtDelUnusedSg m_evtEWDelUnusedSg`

VB: `Public Event m_evtEWDelUnusedSg As dlgtDelUnusedSg`

Value

Type: dlgtDelUnusedSg

CLayoutEngine.m_evtEWEndRender Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtEndRender m_evtEWEndRender`

VB: `Public Event m_evtEWEndRender As dlgtEndRender`

Value

Type: GFXWrpSceneGraph.dlgtEndRender

CLayoutEngine.m_evtEWEEnginPreRenderInfo Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnEnginePreRenderInfo m_evtEWEEnginPreRenderInfo`

VB: `Public Event m_evtEWEEnginPreRenderInfo As dlgtOnEnginePreRenderInfo`

Value

Type: dlgtOnEnginePreRenderInfo

CLayoutEngine.m_evtEWEEnginRenderInfo Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnEngineRenderInfo m_evtEWEEnginRenderInfo`

VB: `Public Event m_evtEWEEnginRenderInfo As dlgtOnEngineRenderInfo`

Value

Type: GFXWrpSceneGraph.dlgtOnEngineRenderInfo

CLayoutEngine.m_evtEWOnBeforeUnloadScene Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnBeforeUnloadScene m_evtEWOnBeforeUnloadScene`

VB: `Public Event m_evtEWOnBeforeUnloadScene As dlgtOnBeforeUnloadScene`

Value

Type: GFXWrpSceneGraph.dlgtOnBeforeUnloadScene

CLayoutEngine.m_evtEWOnSgPrepared Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnSGPrepared m_evtEWOnSgPrepared`

VB: `Public Event m_evtEWOnSgPrepared As dlgtOnSGPrepared`

Value

Type: GFXWrpSceneGraph.dlgtOnSGPrepared

CLayoutEngine.m_evtEWOnSgPreview Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnSgPreview m_evtEWOnSgPreview`

VB: `Public Event m_evtEWOnSgPreview As dlgtOnSgPreview`

Value

Type: GFXWrpSceneGraph.dlgtOnSgPreview

CLayoutEngine.m_evtEWOnSgPreviewForAddins Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnSgPreviewForAddins m_evtEWOnSgPreviewForAddins`

VB: `Public Event m_evtEWOnSgPreviewForAddins As dlgtOnSgPreviewForAddins`

Value

Type: dlgtOnSgPreviewForAddins

CLayoutEngine.m_evtEWOnUnloadScene Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnUnloadScene m_evtEWOnUnloadScene`

VB: `Public Event m_evtEWOnUnloadScene As dlgtOnUnloadScene`

Value

Type: GFXWrpSceneGraph.dlgtOnUnloadScene

CLayoutEngine.m_evtGlobalVariableStatus Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event GlobalVariableStatus m_evtGlobalVariableStatus`

VB: `Public Event m_evtGlobalVariableStatus As GlobalVariableStatus`

Value

Type: GFXWrpSceneGraph.GlobalVariableStatus

CLayoutEngine.m_evtOnDataReady Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnDataReady m_evtOnDataReady`

VB: `Public Event m_evtOnDataReady As dlgtOnDataReady`

Value

Type: GFXWrpSceneGraph.dlgtOnDataReady

CLayoutEngine.m_evtOnEngineInitialized Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnEngineInitialized m_evtOnEngineInitialized`

VB: `Public Event m_evtOnEngineInitialized As dlgtOnEngineInitialized`

Value

Type: GFXWrpSceneGraph.dlgtOnEngineInitialized

CLayoutEngine.m_evtOnRowValuesChanged Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtOnDataRowChanged m_evtOnRowValuesChanged`

VB: `Public Event m_evtOnRowValuesChanged As dlgtOnDataRowChanged`

Value

Type: `GFXWrpSceneGraph.dlgtOnDataRowChanged`

CLayoutEngine.m_evtOnSGPrepareStatus Event

Namespace: `GFXWrpSceneGraph`

Assembly: `GFXWrpSceneGraph` (in `GFXWrpSceneGraph.dll`) Version: 2011

Syntax

C#: `public event dlgtOnSGPrepareStatus m_evtOnSGPrepareStatus`

VB: `Public Event m_evtOnSGPrepareStatus As dlgtOnSGPrepareStatus`

Value

Type: `GFXWrpSceneGraph.dlgtOnSGPrepareStatus`

CLayoutEngine.m_evtOnTableCollectionChanged Event

Namespace: `GFXWrpSceneGraph`

Assembly: `GFXWrpSceneGraph` (in `GFXWrpSceneGraph.dll`) Version: 2011

Syntax

C#: `public event dlgtOnTableChange m_evtOnTableCollectionChanged`

VB: `Public Event m_evtOnTableCollectionChanged As dlgtOnTableChange`

Value

Type: `GFXWrpSceneGraph.dlgtOnTableChange`

CLayoutEngine.m_evtOnTablePageChanged Event

Namespace: `GFXWrpSceneGraph`

Assembly: `GFXWrpSceneGraph` (in `GFXWrpSceneGraph.dll`) Version: 2011

Syntax

C#: `public event dlgtOnTablePageChange m_evtOnTablePageChanged`

VB: `Public Event m_evtOnTablePageChanged As dlgtOnTablePageChange`

Value

Type: GFXWrpSceneGraph.dlgtOnTablePageChange

CLayoutEngine.m_evtRegUnregGlobalAddin Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtRegUnregGlobalAddin m_evtRegUnregGlobalAddin`

VB: `Public Event m_evtRegUnregGlobalAddin As dlgtRegUnregGlobalAddin`

Value

Type: dlgtRegUnregGlobalAddin

CLayoutEngine.m_evtRegUnregGlobalVar Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtRegUnregGlobalVar m_evtRegUnregGlobalVar`

VB: `Public Event m_evtRegUnregGlobalVar As dlgtRegUnregGlobalVar`

Value

Type: dlgtRegUnregGlobalVar

CLayoutEngine.m_evtSSKeyPressE Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtSSKeyPressEvnt m_evtSSKeyPressE`

VB: `Public Event m_evtSSKeyPressE As dlgtSSKeyPressEvnt`

Value

Type: dlgtSSKeyPressEvnt

CLayoutEngine.m_evtSSMouseEvent Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtSSMouseEvent m_evtSSMouseEventE`

VB: `Public Event m_evtSSMouseEventE As dlgtSSMouseEvent`

Value

Type: dlgtSSMouseEvent

CLayoutEngine.m_evtUpdateVariable Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtUpdateVariable m_evtUpdateVariable`

VB: `Public Event m_evtUpdateVariable As dlgtUpdateVariable`

Value

Type: GFXWrpSceneGraph.dlgtUpdateVariable

CLayoutEngine.m_evtVariableChange Event

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#: `public event dlgtVariableChange m_evtVariableChange`

VB: `Public Event m_evtVariableChange As dlgtVariableChange`

Value

Type: GFXWrpSceneGraph.dlgtVariableChange

NodeUpdated

Scene.NodeUpdating Event

This event is raised in WasPI on MouseMove during translate, rotate or scale to any object when mouse left button is pressed.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event NodeUpdatingHandler NodeUpdating`

VB: `Public Event NodeUpdating As NodeUpdatingHandler`

Value

Type: NodeUpdatingHandler

Remarks

Object animatable property should be On.

ITouchEvents.OnDragHover Method

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#:

```
void OnDragHover(
    string sTrgtNode,
    string sSrcNode,
    enumHoverSTATE eState
)
```

VB:

```
Sub OnDragHover (
    sTrgtNode As String,
    sSrcNode As String,
    eState As enumHoverSTATE
)
```

Parameters

sTrgtNode

Type: System.String

sSrcNode

Type: System.String

eState

Type: GFXWrpSceneGraph.enumHoverSTATE

PHStatus

BeforeNodeUpdating Event

This event is raised in Waspl just before user translate, rotate or scale any object when mouse left button is pressed.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event BeforeNodeUpdatingHandler BeforeNodeUpdating`

VB: `Public Event BeforeNodeUpdating As BeforeNodeUpdatingHandler`

Value:

Type: BeeSys.Wasp3D.HostingX.BeforeNodeUpdatingHandler

Remarks:

Object animatable property should be On.

BeforeSceneScrub

This event is raised when user start scrubbing over a scrubbable object in waspi

Namespace: [BeeSys.Wasp3D.HostingX](#)

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event BeforeScrubHandler BeforeSceneScrub`

VB: `Public Event BeforeSceneScrub As BeforeScrubHandler`

Value:

Type: Type: [BeeSys.Wasp3D.HostingX. BeforeScrubHandler](#)

Remarks:

Object animatable property should be On.

CurrentMatSurface

DataReady

This event is triggered when the data in the table is completely loaded in all columns and entire table after executing the specific query or variable update. This event is available in case of a table based data AddIn.

Example: Updating the list of candidates for a state for a specific party.

Generally, the Election database is huge and is full of data tables and entries. The data query might not be executed in time when the animation is called for change. Therefore, the user can use this event to make sure that the data is ready before triggering the animation.

EventBindding

This method is used to attach/detach all the events

Excute Trigger Event

This event is raised when a scene received any predefine or custom trigger. This trigger can be raised from other scene or self scene.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event ExecuteTrggrHandler ExecuteTrigger`

VB: `Public Event ExecuteTrigger As ExecuteTrggrHandler`

Value:

Type: ExecuteTrggrHandler

Remarks:

ExecuteTrigger event deprecated. Use ExecuteTriggerEx instead.

FrameChangeRaise Event

This event is raised when a RaiseEvent action execute through control key.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: public event FrameChangeRaiseHandler FrameChangeRaise

VB: Public Event FrameChangeRaise As FrameChangeRaiseHandler

Value

Type: FrameChangeRaiseHandler

Frame Change Anim Event

This event is raised when a control key execute from animation track.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event FrameChangeAnimHandler FrameChangeAnim`

VB: `Public Event FrameChangeAnim As FrameChangeAnimHandler`

Value

Type: FrameChangeAnimHandler

Live

Map Message Event

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: public event MapMessageHandler MapMessage

VB: Public Event MapMessage As MapMessageHandler

Value

Type: MapMessageHandler

Menu Group Event

Menu Item Exit Event

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event NodeMenuItemExitHandler MenuItemExit`

VB: `Public Event MenuItemExit As NodeMenuItemExitHandler`

Value

Type: NodeMenuItemExitHandler

Menu Item Shuffle Event

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event NodeMenuItemShuffleHandler MenuItemShuffle`

VB: `Public Event MenuItemShuffle As NodeMenuItemShuffleHandler`

Value

Type: NodeMenuItemShuffleHandler

Menu Shuffle Event

This event is raised from Menu Group when more than one menu item complete its animation in one single field.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event MenuShuffleHandler MenuShuffle`

VB: `Public Event MenuShuffle As MenuShuffleHandler`

Value

Type: MenuShuffleHandler

Remarks

When user pass playsteps more than animation offset. MenuShuffle deprecitated, use MenuItemShuffle.

Menu Event

This event is raised from Menu Group when an item reappear to start its animation again.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event MenuEventHandler MenuEvent`

VB: `Public Event MenuEvent As MenuEventHandler`

Value

Type: MenuEventHandler

Remarks

MenuItemEvent Deprecitated, use MenuItemExit instead.

Modifier Prepared Event

Bevel

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event ModiferPreparedHandler ModifierPrepared`

VB: `Public Event ModifierPrepared As ModiferPreparedHandler`

Value

Type: ModiferPreparedHandler

Blast

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event ModiferPreparedHandler ModifierPrepared`

VB: `Public Event ModifierPrepared As ModiferPreparedHandler`

Value

Type: ModiferPreparedHandler

Extrude

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event ModiferPreparedHandler ModifierPrepared`

VB: `Public Event ModifierPrepared As ModiferPreparedHandler`

Value

Type: ModiferPreparedHandler

Lathe

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event ModiferPreparedHandler ModifierPrepared`

VB: `Public Event ModifierPrepared As ModiferPreparedHandler`

Value

Type: ModiferPreparedHandler

Scene

This event is raised from extrude object when it is created asynchronously and ready to render.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event ModiferPreparedHandler ModifierPrepared`

VB: `Public Event ModifierPrepared As ModiferPreparedHandler`

Value

Type: ModiferPreparedHandler

Remarks

It is raised after the polydata of the associated AlFile is changed according to which the Extrude is changed.

Spline Extrude

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event ModiferPreparedHandler ModifierPrepared`

VB: `Public Event ModifierPrepared As ModiferPreparedHandler`

Value

Type: ModiferPreparedHandler

Node Updating

This event is raised in WaspI on MouseMove during translate, rotate or scale to any object when mouse left button is pressed.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

C#: `public event NodeUpdatingHandler NodeUpdating`

VB: `Public Event NodeUpdating As NodeUpdatingHandler`

Value

Type: NodeUpdatingHandler

Remarks

Object animatable property should be On.

Object Prepared

This event is raised when a extrude object or an AI file or a texture file or a video file is loaded asynchronously (file is changed) and ready to render.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event ObjectPreparedHandler ObjectPrepared`

VB: `Public Event ObjectPrepared As ObjectPreparedHandler`

Value

Type: ObjectPreparedHandler

On Drag Hover Method

Namespace: GFXWrpSceneGraph

Assembly: GFXWrpSceneGraph (in GFXWrpSceneGraph.dll) Version: 2011

Syntax

C#:

```
void OnDragHover(
    string sTrgtNode,
    string sSrcNode,
    enumHoverSTATE eState
)
```

VB:

```
Sub OnDragHover (
    sTrgtNode As String,
    sSrcNode As String,
    eState As enumHoverSTATE
)
```

Parameters

`sTrgtNode`

Type: System.String

`sSrcNode`

Type: System.String

`eState`

Type: GFXWrpSceneGraph.enumHoverSTATE

On Frame Change Method

This event is raised when a control key execute in scene except from animation track.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#:

```
public void OnFrameChange(  
    int trackID,  
    int currentFrame  
)
```

VB:

```
Public Sub OnFrameChange (  
    trackID As Integer,  
    currentFrame As Integer  
)
```

Parameters

trackID

Type: System.Int32

currentFrame

Type: System.Int32

On Gesture

This event is raised when a gesture execute in scene which having a RaiseEvent action on it

Namespace: [BeeSys.Wasp3D.HostingX](#)

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

C#: public event GestureResponseHandler

VB: Public Event GestureResponse As GestureResponseHandler

On Named Event

This event is raised when a NamedEvent is received in scene. This received named event can be from self scene or from other scene

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

C#: public event NamedEventHandler Named

VB: Public Event Named As NamedEventHandler

On Page Changed

This event is raised when a current page index change in datasource addin. When user perform record move action or do next,previous of datatable addin then this event is occur.

C#: public event TablePageChangeHandler PageChanged

VB: Event PageChanged As TablePageChangeHandler

Value

Type: TablePageChangeHandler

On Scene Cue

This event is raised when F9 button pressed from sting-client after scene load or Cue button pressed in designer or SceneCue action execute.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

C#: public event SceneCueHandler SceneCueEvent

VB: Public Event SceneCueEvent As SceneCueHandler

Value

Type: SceneCueHandler

Remarks

It is used to reset scene animation.

On Scene Prepare

This event is raised just after successful loading of the scene.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

C#: public event PreparedHandler ScenePrepared

VB: Public Event ScenePrepared As PreparedHandler

Value

Type: PreparedHandler

On Scene Preview

This method is used to get the preview image of the scene

On Scene Unload

Executes the list of actions when a scene is unloaded.

Example: Playing another graphic automatically when the current is unloaded.

On Trigger

Executes the list of actions when a trigger is received in a scene.

Gives access to a list of Predefined, Custom and Local triggers.

- Predefined Triggers: Comes default with the designer
- Custom Triggers: Can be added to the list by clicking on '+' icon and deleted by using the '-' icon.
Custom triggers are available in all the scenes. These variables are useful when user wants to do actions in multiple scenes when a trigger is raised. For e.g. moving a ticker up or down in case of an advertisement along with hiding the news flash or name straps etc.
- Local Triggers: Are added to this scene only they are not available in other scenes. These triggers are restricted to the scene only.



Trigger ID Custom Triggers are available between 1 and 1000 and from 1200 to 2000 in Local Triggers

On Variable Changed

Actions can be added and executed every time the value of an addin changes.

Example: Calling to display different data for each player in case of a game each time the name/country of a player is changed.

Page End Event

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event NodePageHandler PageEnd`

VB: `Public Event PageEnd As NodePageHandler`

Value

Type: NodePageHandler

Render State Event

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event RenderStateHandler RenderState`

VB: `Public Event RenderState As RenderStateHandler`

Value

Type: RenderStateHandler

Row Values Method

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#:

```
public bool _RowValue(
    XmlNode rowNode,
    ref int row,
    ref string value
)
```

VB:

```
Public Function _RowValue (
    rowNode As XmlNode,
    ByRef row As Integer,
    ByRef value As String
) As Boolean
```

Parameters

`rowNode`

Type: System.Xml.XmlNode

`row`

Type: System.Int32

`value`

Type: System.String

Return Value

Type: Boolean

Scene Next Graphics

It triggers the next instance in the sting client rundown to Load (if the player is empty) or Play (if the player is loaded) on the selected Sting Server.

Scene Page Out Event

This command sends the Page-Out trigger to the Scene. Page-Out is commonly used with Page Controller object. It checks if there are no more pages in the Data Pages controller and if loop command is turned off, it will exit from the page animation loop and trigger the default controller.

Target – Allows user to select self (where the action is applied) or an object with additional options of selecting its animation track, parametric properties to which the animation set will be applied.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event PageOutHandler ScenePageOut`

VB: `Public Event ScenePageOut As PageOutHandler`

Value

Type: BeeSys.Wasp3D.HostingX.PageOutHandler

Remarks

ScenePageOut event deprecated. Use PageEnd event instead of ScenePageOut.

Scene Play Complete

This is a notification event to unload the scene. Its can be raised by PlayComplete action.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event PlayCompleteHandler ScenePlayComplete`

VB: `Public Event ScenePlayComplete As PlayCompleteHandler`

Value

Type: PlayCompleteHandler

Scene Scrub Event

This event is raised in waspi when scrubbing an object. This event raise only for scrubbable objects.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event ScrubHandler SceneScrub`

VB: `Public Event SceneScrub As ScrubHandler`

Value

Type: ScrubHandler

Remarks

Object scrubbable property should be On.

Scene.BeforeSceneScrub Event

This event is raised when user start scrubbing over a scrubbable object in waspi

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

C#: `public event BeforeScrubHandler BeforeSceneScrub`

VB: `Public Event BeforeSceneScrub As BeforeScrubHandler`

Value: Type: BeeSys.Wasp3D.HostingX.BeforeScrubHandler

Scroller Event

This event is raised by scroller object for multiple stages of scroller group like HEAD_ENTRY, TAIL_ENTRY, HEAD_EXIT.....

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event SGScrollerHandler Scroller`

VB: `Public Event Scroller As SGScrollerHandler`

Value

Type: SGScrollerHandler

Remarks

ScrollerEvent deprecitated, use Scroller instead.

Scroller

Status Event

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event StatusX Status`

VB: `Public Event Status As StatusX`

Value

Type: StatusX

Table Collection Changed Event

This event is raised when a datatable collection is changed in dataset addin. If a table added or deleted from dataset addin then this event is occurs.

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event TableChangeHandler TableCollectionChanged`

VB: `Public Event TableCollectionChanged As TableChangeHandler`

Value

Type: TableChangeHandler

Video Event

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event NodeVideoHandler VideoEvent`

VB: `Public Event VideoEvent As NodeVideoHandler`

Value

Type: NodeVideoHandler

Video Time Code Event

This event is raised when video time code changed. It denote video frame time in time code

Namespace: BeeSys.Wasp3D.HostingX

Assembly: BeeSys.Wasp3D.Hosting.SceneGraph (in BeeSys.Wasp3D.Hosting.SceneGraph.dll) Version: 2011

Syntax

C#: `public event VideoTimecodeHandler VideoTimeCode`

VB: `Public Event VideoTimeCode As VideoTimecodeHandler`

Value

Type: `VideoTimecodeHandler`

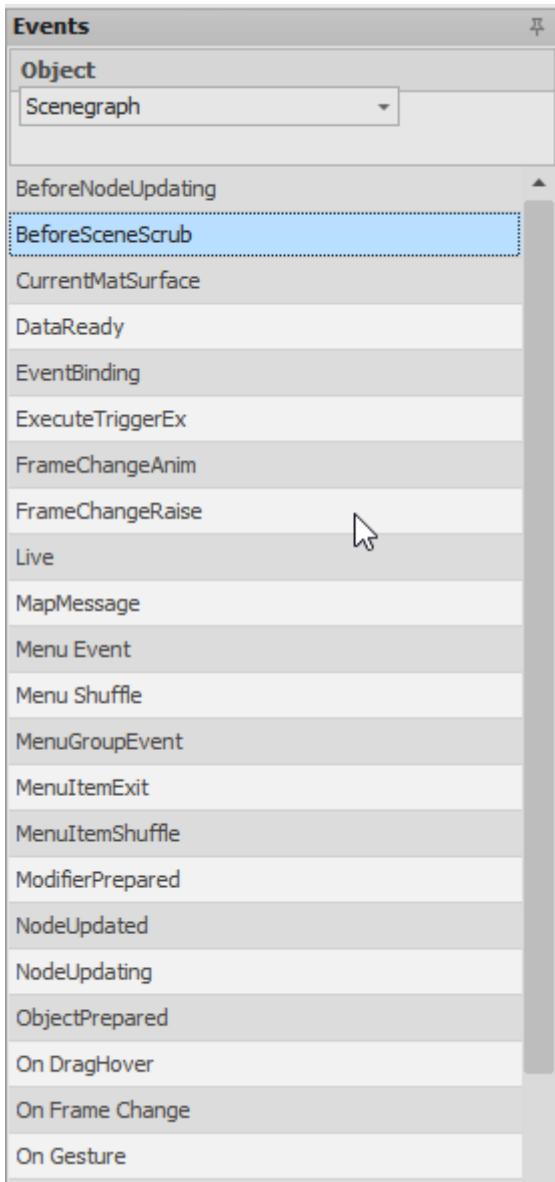
Video

Error List

Shows a list of errors that arise during the compilation of code.

Events

Scene Graph



BeforeNodeUpdating:

BeforeSceneScrub:

CurrentMatSurface:

DataReady:

EventBindding:

Excute TriggerEx;

Frame Change Anim:

Frame Change Raise:

Live:

Map Message:

Menu:

Event:

Menu Shuffle:

Menu Group Event:

Menu Item Exit:

Menu Item Shuffle:

Modifier Prepared:

Node Updating:

Object Prepared:

On Drag Hover:

On Frame Change:

On Gesture:

On Named Event:

On Page Changed:

On Scene Cue:

On Scene Prepare:

On Scene Preview:

On Scene Unload:

On Trigger:

On Variable Changed:

Page End:

RenderState:

RowValuesChanged:

SceneNextGraphics:

ScenepageOut:

ScenePlayComplete:

SceneScrub:

Scroller:

ScrollerEvent:

Status:

TableCollectionChanged:

Video:

VideoEvent:

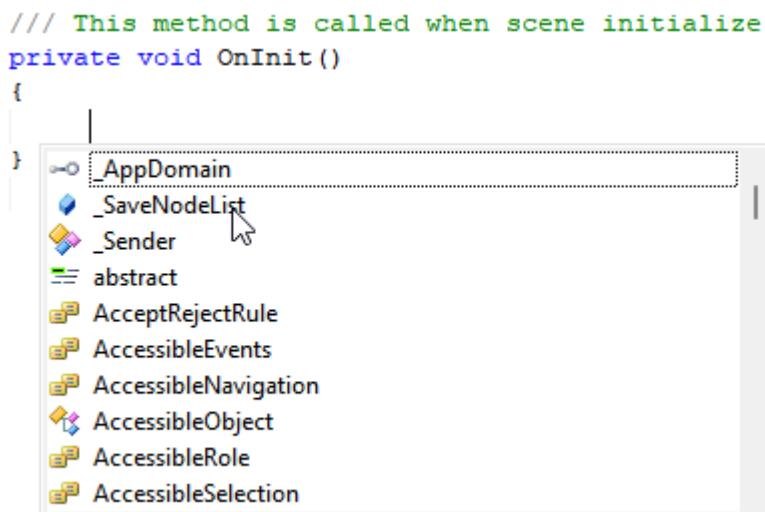
VideoTmeCode:

Scene Graph

A placeholder where you can write code.

Intellisence

Class Drop-Down and Method Drop-Down - Displays a list of Classes and Methods available in selected class



Solution Explorer

Contains a list of Default Class Libraries and Classes that are being used to write code.

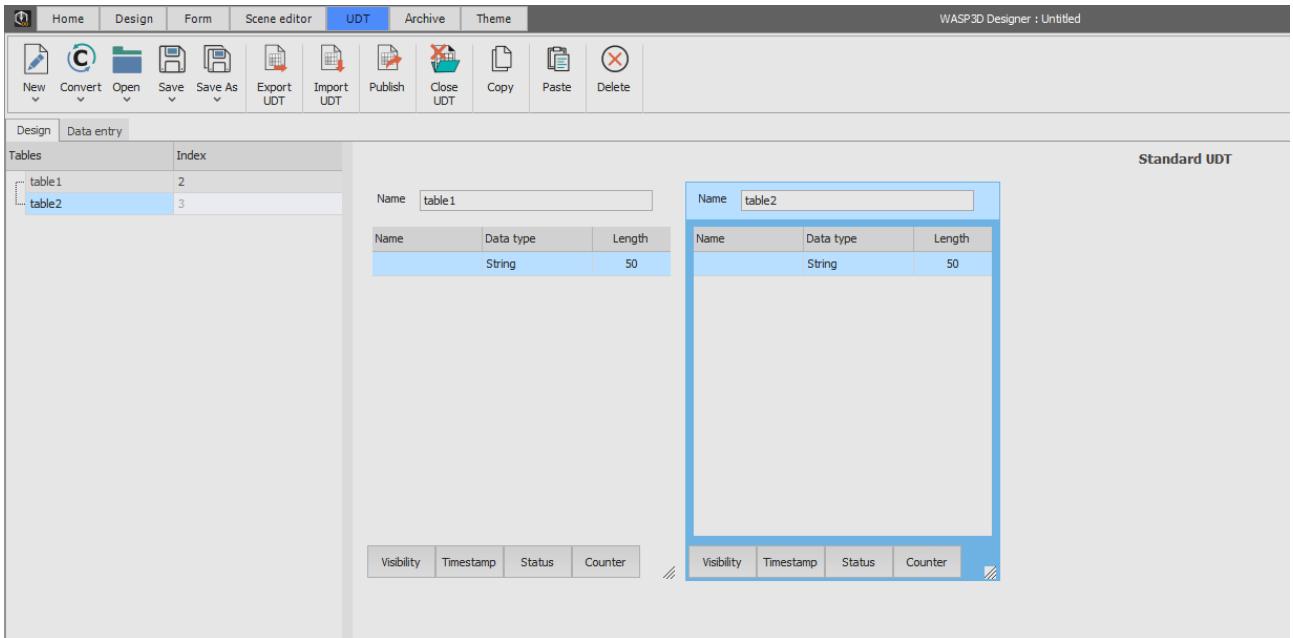
Tools

Contains the icons required for Scene Editor.

UDT

1. To add a new table at the root level, press insert key or right-click on the Name bar of the table hierarchy column and select Add Table.

2. To add a child table, select the parent and press Insert key or right-click and choose Add Table.
3. To rename a table, select and press F2.
4. To delete a table, select and press Delete key or right-click and choose Delete Table.



Data Types in UDT

- Integer: Variables that have whole numbers as their value. These are variables, which occupy 4 bytes in memory and can take both positive and negative values.
- String: String is a data type that represents sequences of characters.
- Float: Values, which are not integral, are stored as floating point numbers.
A floating-point number can be expressed as a decimal value such as 112.5, or with an exponent such as 1.125E2 where the decimal part is multiplied by the power of 10 specified after the E (for Exponent). A floating point constant must contain a decimal point, or an exponent, or both. If you write neither, you have an integer.
- Boolean: A data type, having two values (denoted as true and false)
- Float 2: Describes a two-component vector including 2 float structures (FLOAT X, and FLOAT Y)
- Float 3: Describes a two-component vector including 3 float structures (FLOAT X, FLOAT Y and FLOAT Z)
- Float 4: Describes a two-component vector including 4 float structures (FLOAT X, FLOAT Y, FLOAT Z and FLOAT W)
- Color: Describes a Four-component vector including 4 floats representing Red, Green, Blue, Alpha.
- Date: Select to input Date in the specified format.
- AI: Select to browse the AI file.
- XFile: Select to browse the .X file.
- NurbFile: Select to browse the FBX or .3DM file.
- Environment: Select to browse the DDS environmental texture file.
- Flash: Select to browse the SWF file.
- PDF: Select to browse the PDF file.
- SHP: Select to browse the SHP file.

- Image: Select to browse the Standard image formats live .jpg, .png, etc.
- Video: Select to browse the video or audio files like .avi, .mov, .wmv, .mp3, etc.
- Datasource: Select Data Source to bind the data with some other column of the table within the same UDT.
- Data Columns can be linked to different tables or columns in the same UDT. This option helps user to fetch data from other tables based on the table selection in "Display in Graphic" and "Display in Table" parameters. This helps user to save time and energy by not requiring entering data multiple times for various graphic requirements.

GoTo Data Binding option available in right side panel, select the table with which you want to bind the data. "Display in Graphics", will show the data to be wired in the graphic & Display in Table will show the data in the table. For e.g. The user selects India and the path of the Flag of India (X:/wasp3d/textures/flags/india.jpeg) will be fetched from the column defined in the "Display in Graphic".

Archive

Archive Tab includes all the tools needed to archive or extract .arc, .wcz, .arcx Files.



This Option allows us to choose between Archive or Extract files.



This Panel appears when needed to Archive Files. This provides options to choose between a Scene, Template, Formpool (.WSF), FormTemplate(.WFT), Playlist, Program. Once selected, click OK to open specific Panels associated with the options above.

Archive – Scenes

This gives us the option to archive Scenes from the Local machine to a desired destination.

Archive – FormTemplate(.WST)

This gives us the option to archive FormTemplates from the Local machine to a desired destination.

Archive – Templates

This gives us the option to archive Templates from the Server to a desired destination.

Archive – Playlist

This gives us the option to archive Playlist from the Server to a desired destination.

Archive – FormPool(.WSF)

This gives us the option to archive FormPool from the Server to a desired destination.

Archive – Program

This gives us the option to archive Program from the Server to a desired destination.

Extract



Source to be Extract:

Option to pick Arc files from a desired Path to extract the files.

Extract Options:

1. Scene: Option to extract the Scene from a separate desired Path.
2. Textures: Option to extract the textures from a separate desired Path.
3. Meshes: Option to extract the Meshes from a separate desired Path.
4. Shapes: Option to extract the Shapes from a separate desired Path.
5. **Data:** Option to extract the Data from a separate desired Path.

Form

Theme

Introduction to WASP3D Solution

WASP3D is a workflow based real-time 3D Graphics solution, complete with content management and its on-air delivery. Ideally suited for Television Stations, Live Graphics Automation Developers, and Post-Production Houses with multiple points of graphics delivery (*On-Air Graphics, NLE stations, Automated Graphics, Downstream Tickers, etc.*). [WASP3D workflow](#) has been designed specifically for On-Air Graphics production in television broadcast stations. WASP3D comes with a touch screen application WASPi Mimosa that makes presentations much more impressive, interactive and easy. Be it elections, business, live events, sports or weather, anchors and reporters can now demonstrate information like weather updates, vote swings, change in stock prices, currency fluctuations and more with a simple touch of a flat panel display or video wall.

WASPi Mimosa gives the user (anchor) extreme flexibility and total control over the graphics system. So, instead of using a mouse click, you are using your hands for pointing, clicking and causing an interaction on the screen. Gestures & touches; stretches & spins; zoom & pan; you can play around with the graphics at the touch of your fingers.

WASP3D comes with a unique UBT (Unified Basic Template) feature that largely helps in speeding up the workflow by allowing graphic artists to create design interface in the Drone Designer module. The solution supports both High Definition (HD) and Standard Definition (SD) video formats.

WASP3D Modules

Drone Designer

Drone Designer is the graphics creation module of WASP3D workflow system. Highly intuitive interface with a low learning curve, Drone Designer allows a graphic artist to generate, manipulate, animate and visualize with 3D objects, images and videos with great ease. The solution is equipped with ready to use design effects such as refraction, reflection, X-ray, real-time cloth, text effect, tile effect and many more. Users can create data-tags of object parameters to enable external data manipulation via the Data Buzz or ActiveX MOS control.

Comb Builder

Comb Builder integrates easy drag and drop controls for creating custom user interface templates for data entry forms in WASP3D scenes. The controls added to a form can simply be associated to WASP3D data tags to be available for data entry. Comb Builder is powered by Microsoft VSTA© which utilizes a common, user friendly, easy coding method to create a custom user interface for data entry applications. Comb Builder is useful in events where complex data graphics, logic and analysis are mandatory in meeting the requirements of elections, sports, or business news programming.

Comb Builder includes a comprehensive visual programming IDE, supporting C#.NET and VB.NET. Developers can create complex data driven interfaces using Comb Builder. The user can connect to databases, wire-feeds, RSS feeds and data gateways to build automation support for data entry. In

addition, it offers a host of options like publishing WASP3D templates, creating custom playlist, shotbox SDK application, WASPi Mimosa code behind, better GUI, IDE integrated help and many more controls & features available in .Net 2.0/3.5/4.0.

Hive Manager

HiveManager is the core application of the WASP3D networked workflow. It provides content management and delivery of data and media objects to all WASP3D clients. It consists of a Web Server, a Database Server and optionally a Storage Server.

The following application modules are available for administrators to manage the WASP3D workflow:

- Kernel Controller: The WASP3D Kernel Controller server manages, maintains and delivers data to all the WASP clients across the network. It is the heart of the WASP3D networked solution. The Kernel Controller is built on top of the high performance Microsoft SQL© based database enabling content sharing across the WASP network. The Kernel Controller also ensures that concurrent data is available to all WASP Data Buzz clients across the network.
- Hive Manager: This is an extensive application for data management of Playlists. Programs, Playlists and Themes can be created through this central administration Playlist Management software. The Hive Manager provides the flexibility to mix-n-match multiple Playlists to create a larger Playlist.

WASP Asset eXchange

WASP Asset eXchange (WAX) module has been designed for sharing, managing and distributing the WASP media assets across local and remote locations of the client. The eXchange tool easily gets integrated with most of the newsroom systems enabling the user to access images and graphic elements and quickly create graphics using templates using its drag and drop feature into a news rundown. eXchange also provides users with the flexibility to search assets based on metadata, displaying a thumbnail view of media assets for enhanced search and navigation.

Data Buzz

Data Buzz is the data acquisition client application. The user interface resides within the WASP3D scene created in the Drone Designer and is available for the non-design personnel to enter/modify the data fields and post them to a playlist. Data Buzz is an operational template used for creating playlists, programs and custom themes. It provides News Desk Editor and/or Journalist the flexibility to manually populate the data entry form with data elements such as name, headlines, breaking news etc. including an available real-time data preview option. Alternatively, custom broadcast applications involving complex data graphics can be created through the data entry interface in the Comb Builder module of WASP3D.

An ActiveX MOS version of the data entry client is available for integration with the newsroom system.

Sting On-Air Playout (Client and Engine)

The STING On-Air is the Graphics Delivery (Playout) application. Designed for performance, the Sting On-Air delivers the graphics in two modes: automated triggering of graphics for unattended operations and manual triggering for production control based delivery. In the manual mode, the Playout

component recognizes the user actions defined (pause-points, continue, play etc.) and waits for the user decision to proceed.

Sting On-Air consists of the Sting On-Air Engine and the Sting On-Air Client. The Engine is the real-time renderer and provides the SDI video I/O (Key and Fill) along with CVBS output (fill) for preview. The Sting On-Air Client is the controlling application of the Engine. When multiple Engines are available on the WASP network, users can choose any of the Engines to trigger the Graphics output.

WASPi Mimosa

WASPi Mimosa is an application that allows designers to create interactive 3D graphics that can be displayed on a touch screen and can be controlled by the anchor/presenter. It allows the user to present the content (zoom, pan, stretch, spin) at the touch of fingers. For creating interactive content, designers can create various objects and animation inside the scene and mark them as 'Interactive Enabled' to allow various objects related operations to be performed on it from the WASPi Mimosa application. The designer can also bind various activities like an animation play or pause to be performed when a specific gesture is performed on a specified object.

