



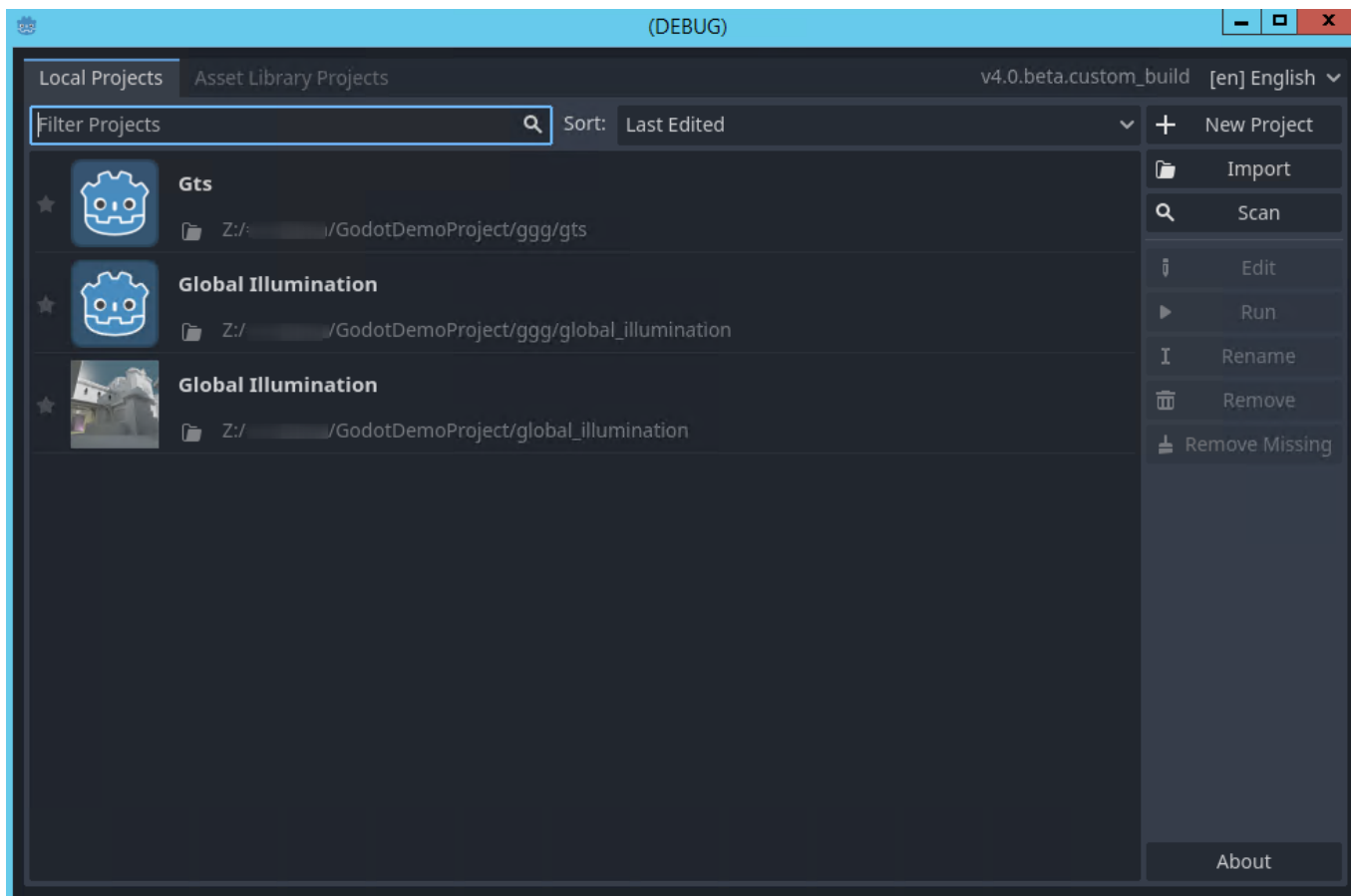
# Godot Project Editor

Server Module **Project Editor**

 The description is valid for software version [2.9.123\\_99](#) and later.

Godot Project Editor is supplied with the Skylark SL NEO software kit. You can run the editor through Start→Programs→SL NEO Media Platform→Godot Engine or directly: C:\Program Files (x86)\SL NEO Media Platform\godot.exe. If you use custom Skylark nodes in your projects, use only the editor that comes with the software kit.

 Do not use editors from the official Godot website or the web-version of the editor, as they do not support integration with Skylark SL NEO Media Platform by default.



## Terms

- **Resource Pack** is a collection of scenes, textures, scripts and other files packed into a .zip or

.pck file. These files can be loaded into the Godot Engine module.

- **Node** - object in the project tree.
- **Scene** - a group of objects represented by a tree structure. A scene can be a preset (prefab). Scenes can be nested in other scenes.
- **Head Scene** - main scene specified in the project parameters as the initial scene.
- **Script** - block of GDScript code linked to an object.
- **Visibility** - Objects lower in the tree overlap with objects higher in the tree
- **Z-Index** - change the drawing position without changing the sequence of objects in the tree.

## Hotkey Combinations

[Godot Editor \(Hot Keys\)](#).

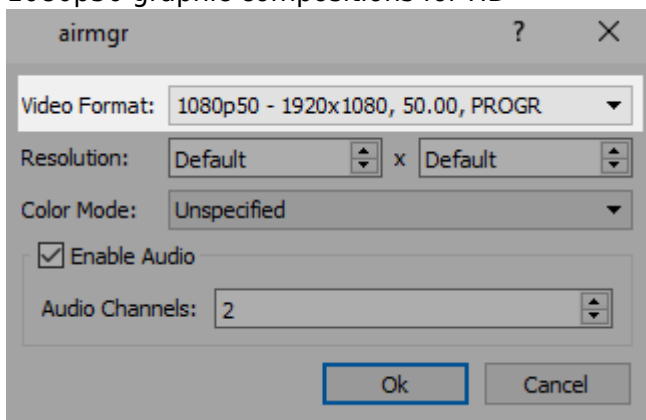
## Tips

[Tips for Working with Godot Editor](#).

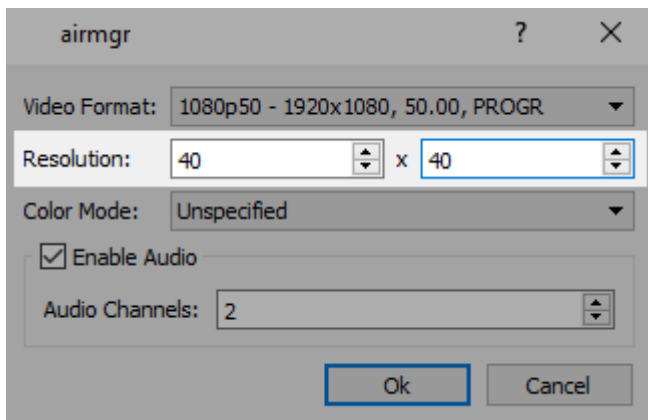
## Optimization

Developing Godot Engine projects, you can stick to a few simple rules to optimize the use of your platform’s resources and get the best results:

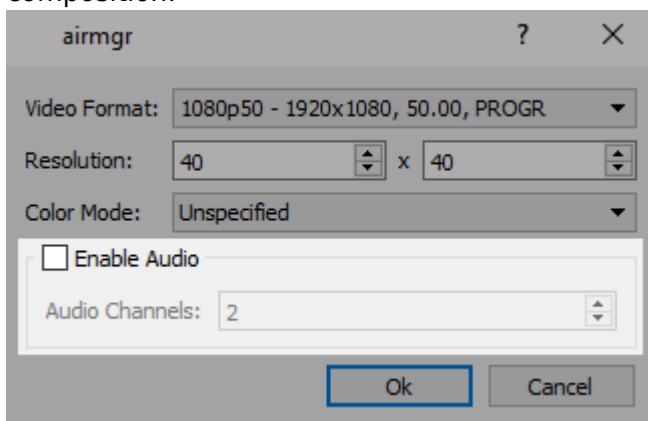
- Create graphic compositions (SLG) and their parts for playout in the Godot Engine player in 50 fps progressive format when maximum quality is required. This will minimize the number of conversions due to Godot Engine performing rendering in this format. For example, create 1080p50 graphic compositions for HD



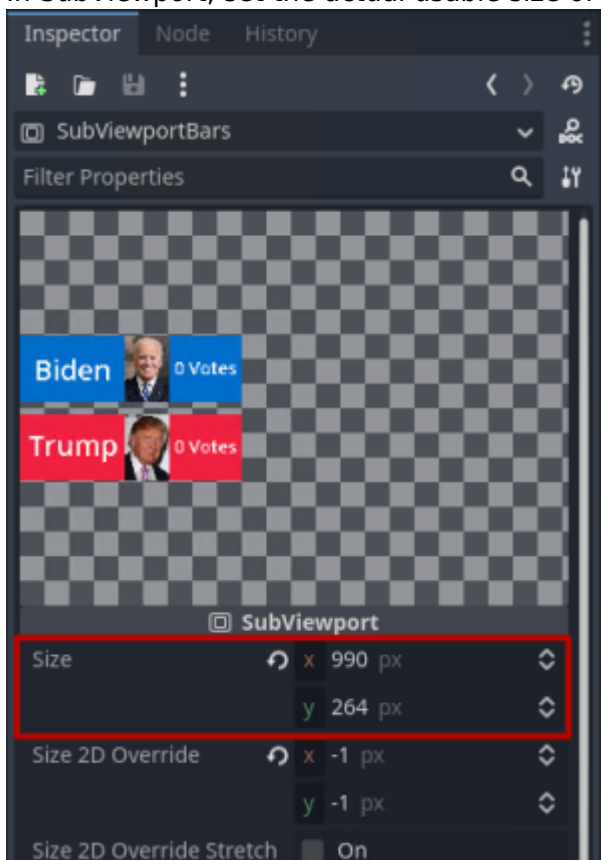
- Create graphic compositions (SLG) and their parts for playout in the Godot Engine player in 25 fps interlaced format, when maximum optimization is required to reduce load. For example, create 1080i50 graphic compositions for HD.
- Create graphic compositions (SLG) with resolution corresponding to the usable image size only. Redundant areas with transparency carry no useful information, while actively utilizing system resources. For instance, to place a 40px/40px logo, set Resolution to 40x40:



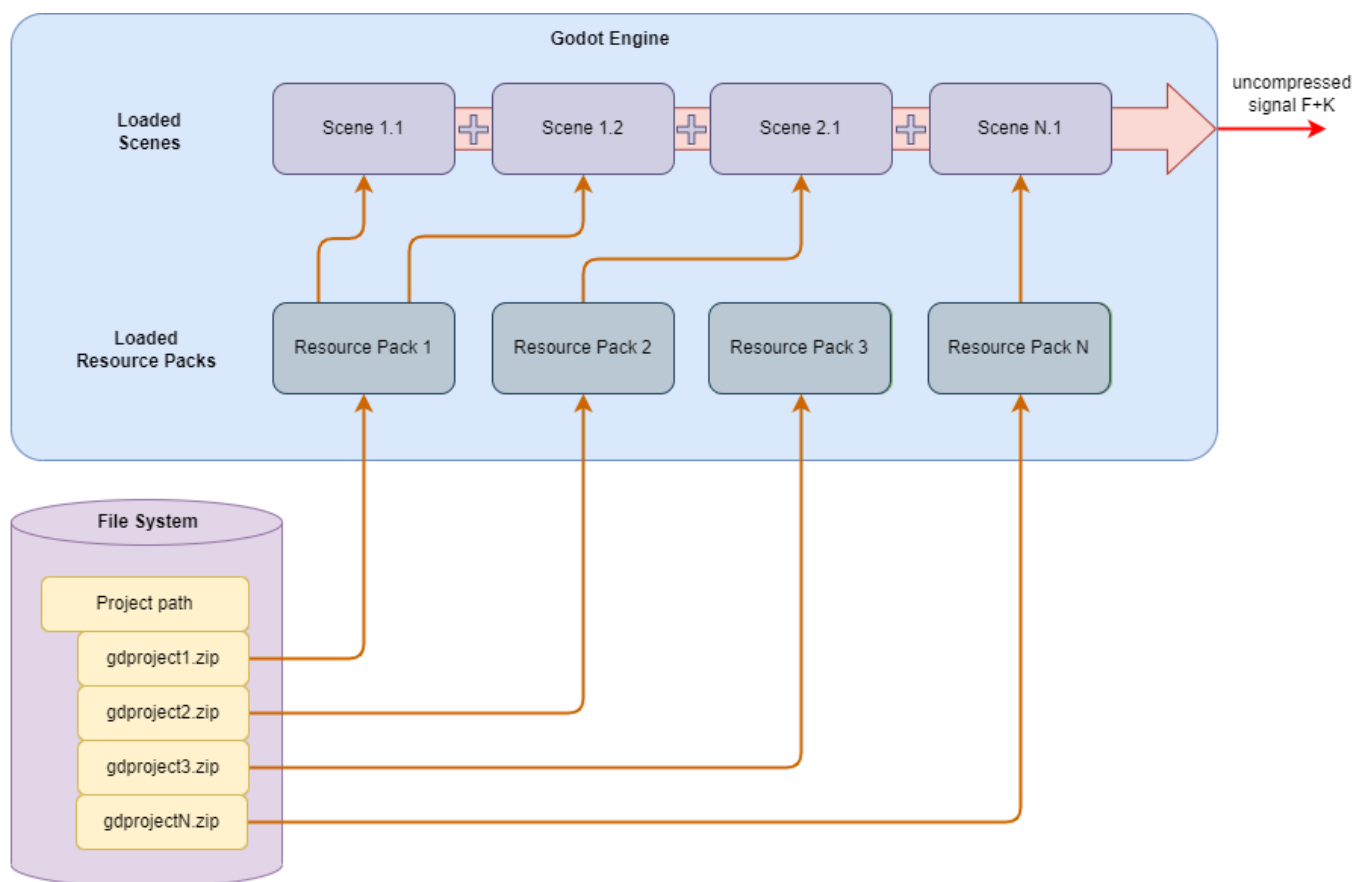
- If the graphic composition (SLG) does not contain sound, disable its use when creating the composition:



- If the SLVideoStreamPlayer node has a clip loaded and played, resources are consumed even if the node is not currently visible on screen. When possible, pause playback when the object is hidden. Use Property→paused for that.
- In SubViewport, set the actual usable size of the loaded scene/composition.



# Creating Custom Scenes



In order to load multiple scenes by different users to Godot Engine without file names overlapping, users are expected to follow a certain naming and object placement structure:

- Custom projects created in the editor require creating a folder with a certain name (e.g. “test”) and placing all files in it.
- The main scene of the project must be named the same as the folder. That is, the “test.tscn” scene must be created in the “test” folder, which will be the root of the custom project.

For example, there is Project path=Z:\GodotProjects and a D:\User\Godot working folder with custom projects.

The user creates a new project in the D:\User\Godot folder. The editor will automatically create the following files:

- D:\User\Godot\icon.png
- D:\User\Godot\icon.png.import
- D:\User\Godot\project.godot

Then, create a new folder in the editor (e.g. test) and place the rest of the files in there:

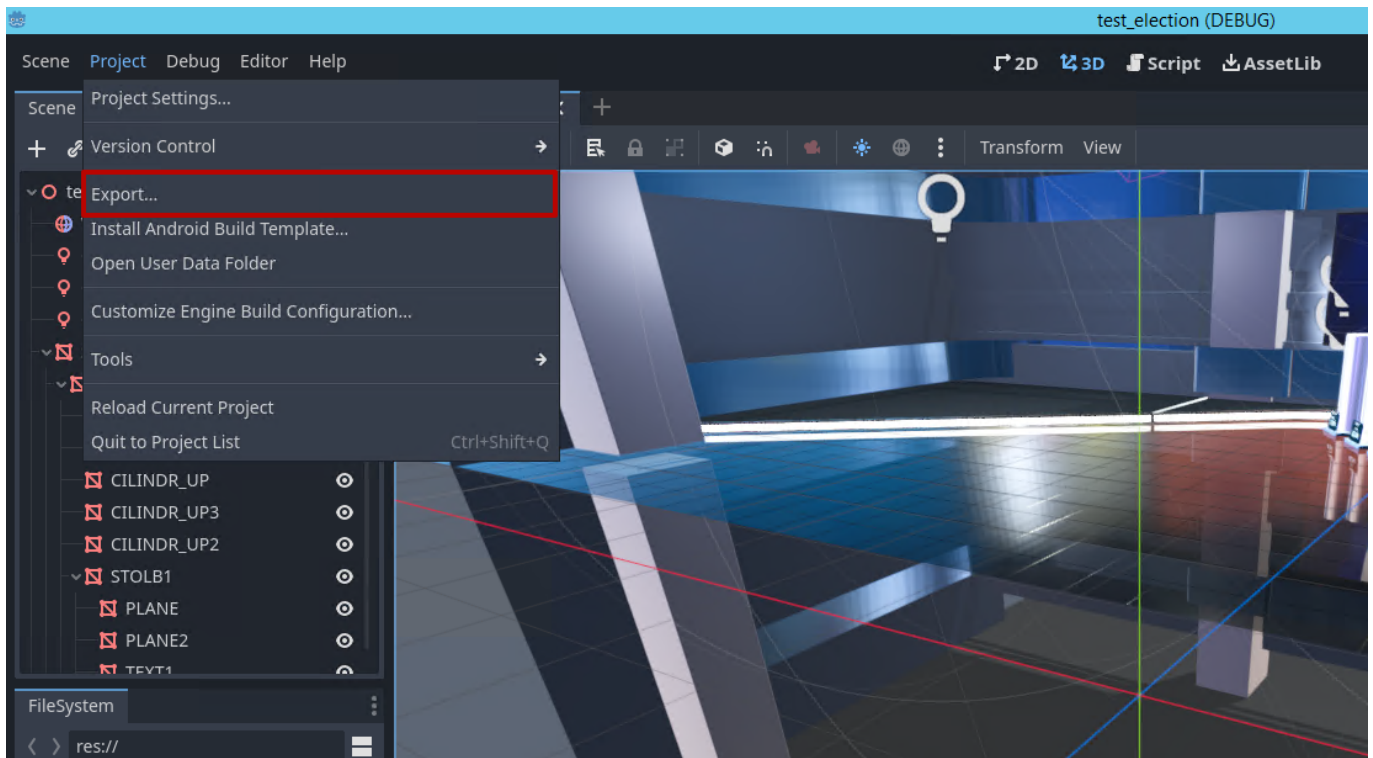
- D:\User\Godot\test\anim.gd
- D:\User\Godot\test\other\_label.tscn
- D:\User\Godot\test\PG1.sls
- D:\User\Godot\test\test.gd
- D:\User\Godot\test\test.tscn (main scene of the project)

In the editor, export the project to a resource pack in \*.zip format.

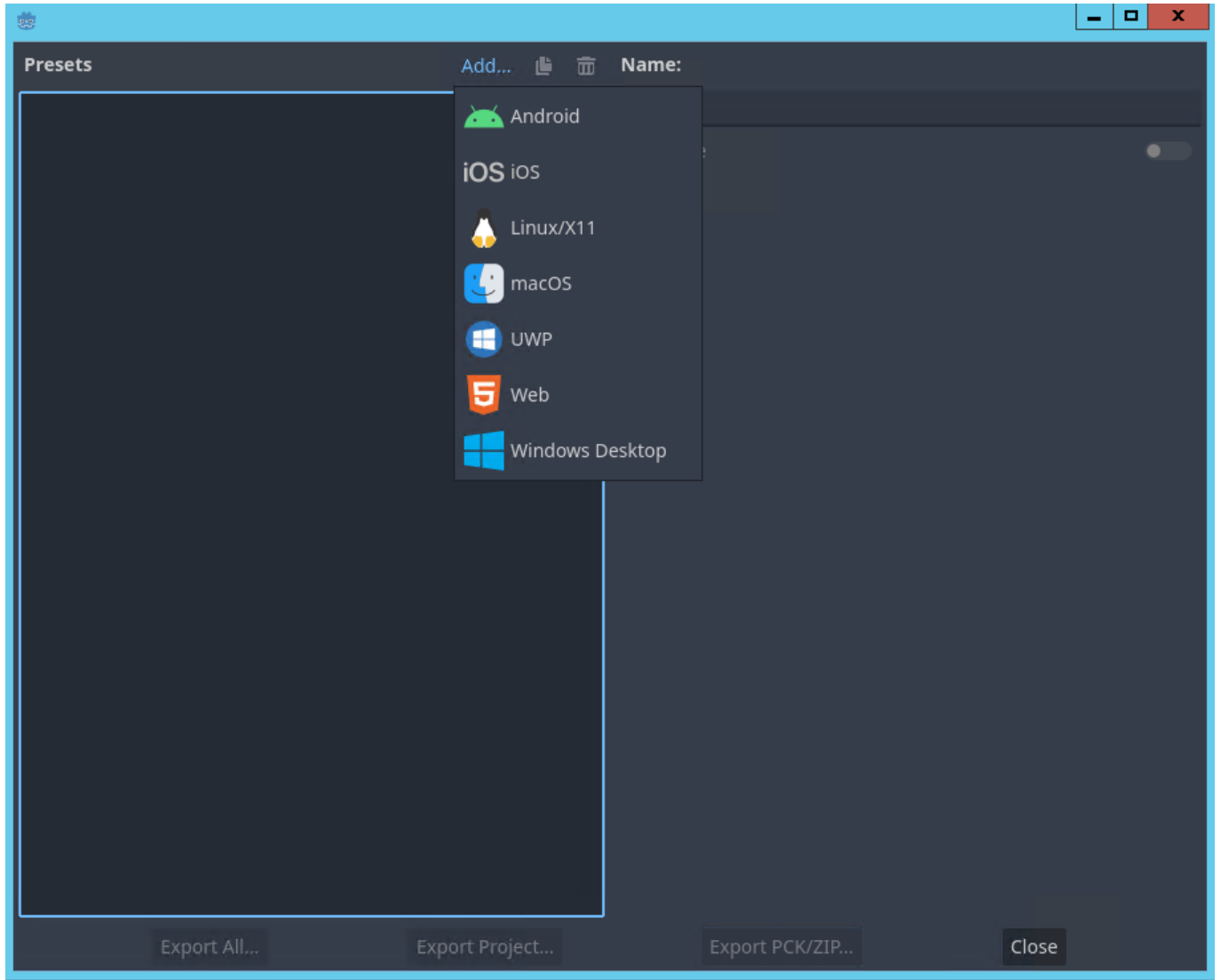
You can place the created resource pack in the Z:\GodotProjects folder, and then load the needed scene by its name in the control panel: Administrator Control Panel→Status→GodotEngine\_N→Load scene=test.

## Export Project to Resource Pack

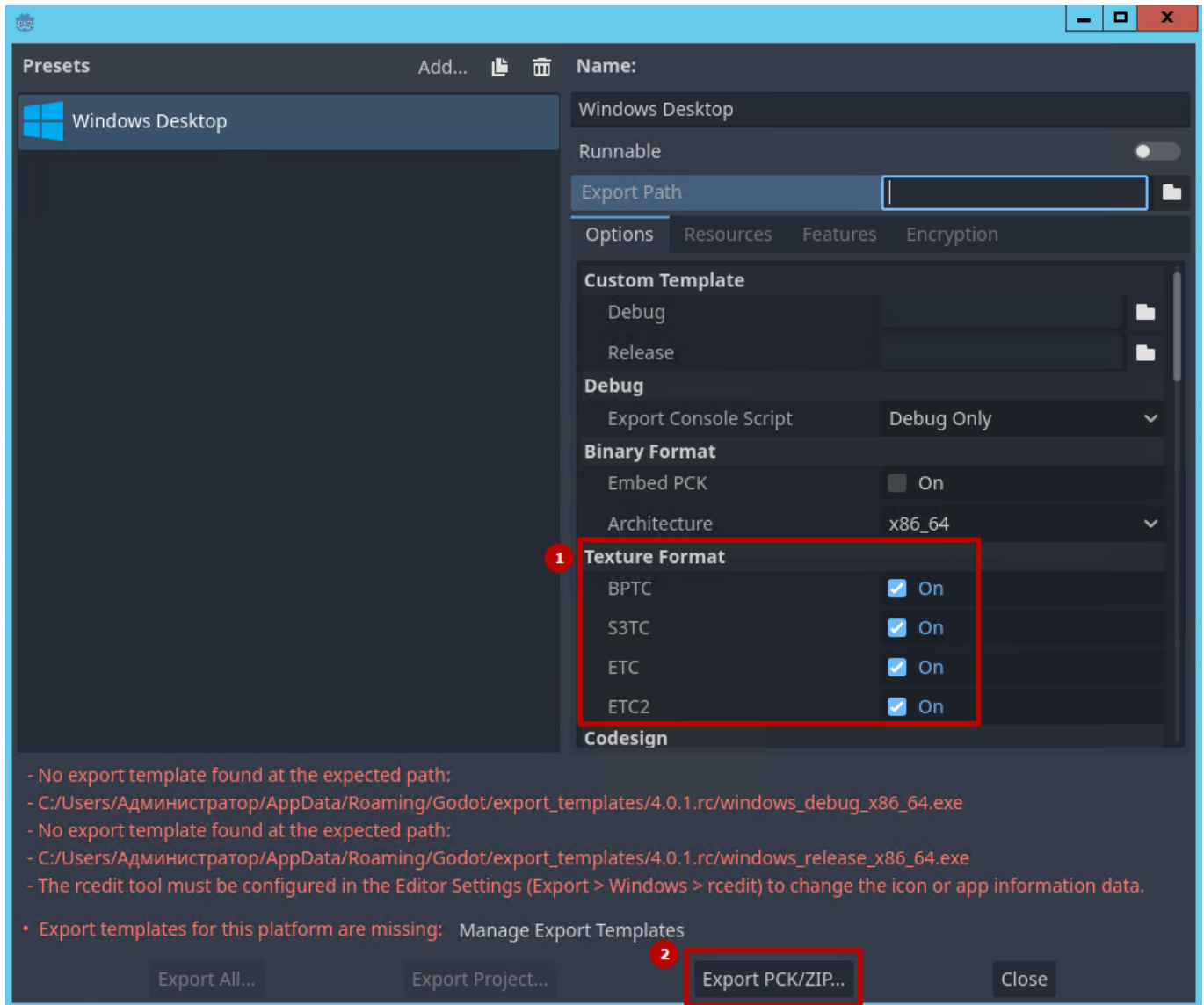
Before loading projects into the Godot module, they must be exported from the editor to the resource pack.



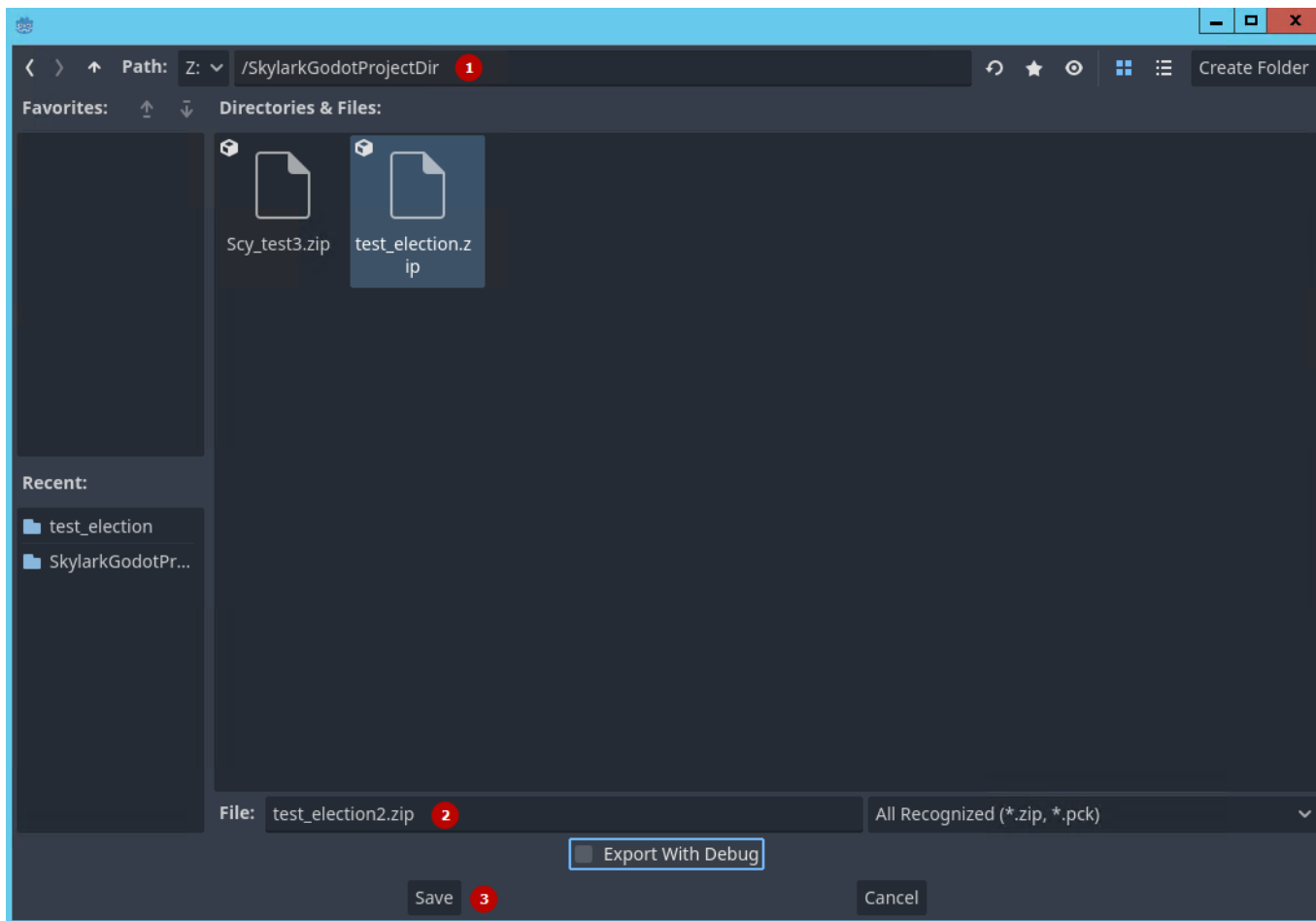
Select the project export option on the menu: Godot Editor→Project Menu→Export....



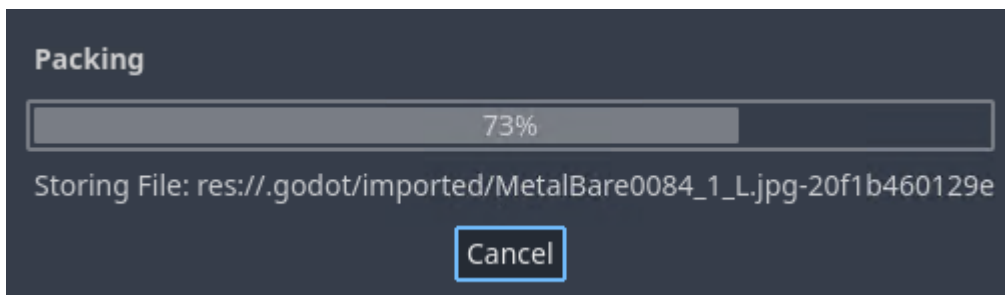
Add a Windows Desktop upload the preset via Add....



Check all texture formats and press Export PCK/ZIP....



- Set the upload folder in the Path menu.
- Specify the uploaded file's name in the File menu. The file name must coincide with the project's main scene name.

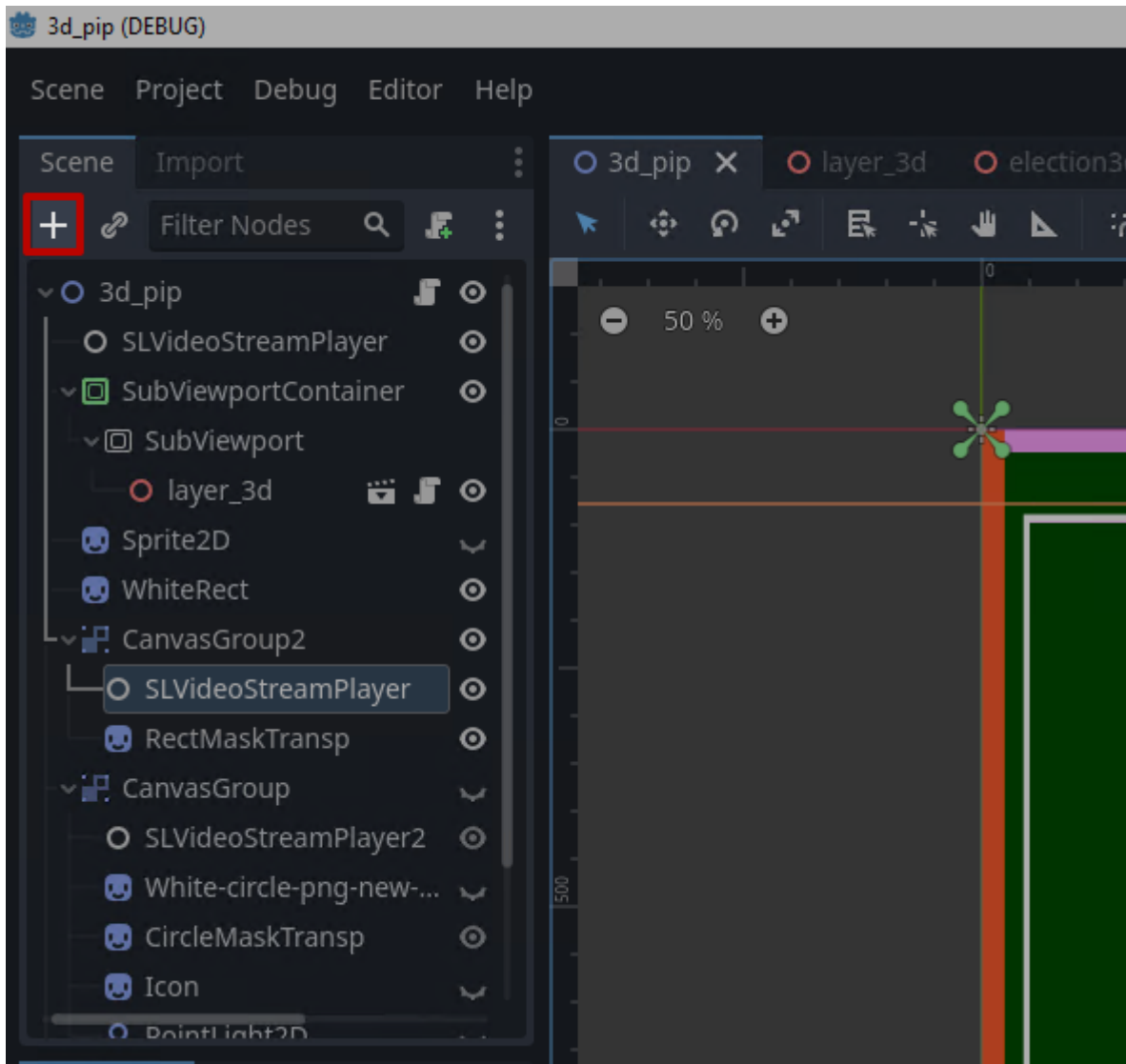


Export of project files will take some time.

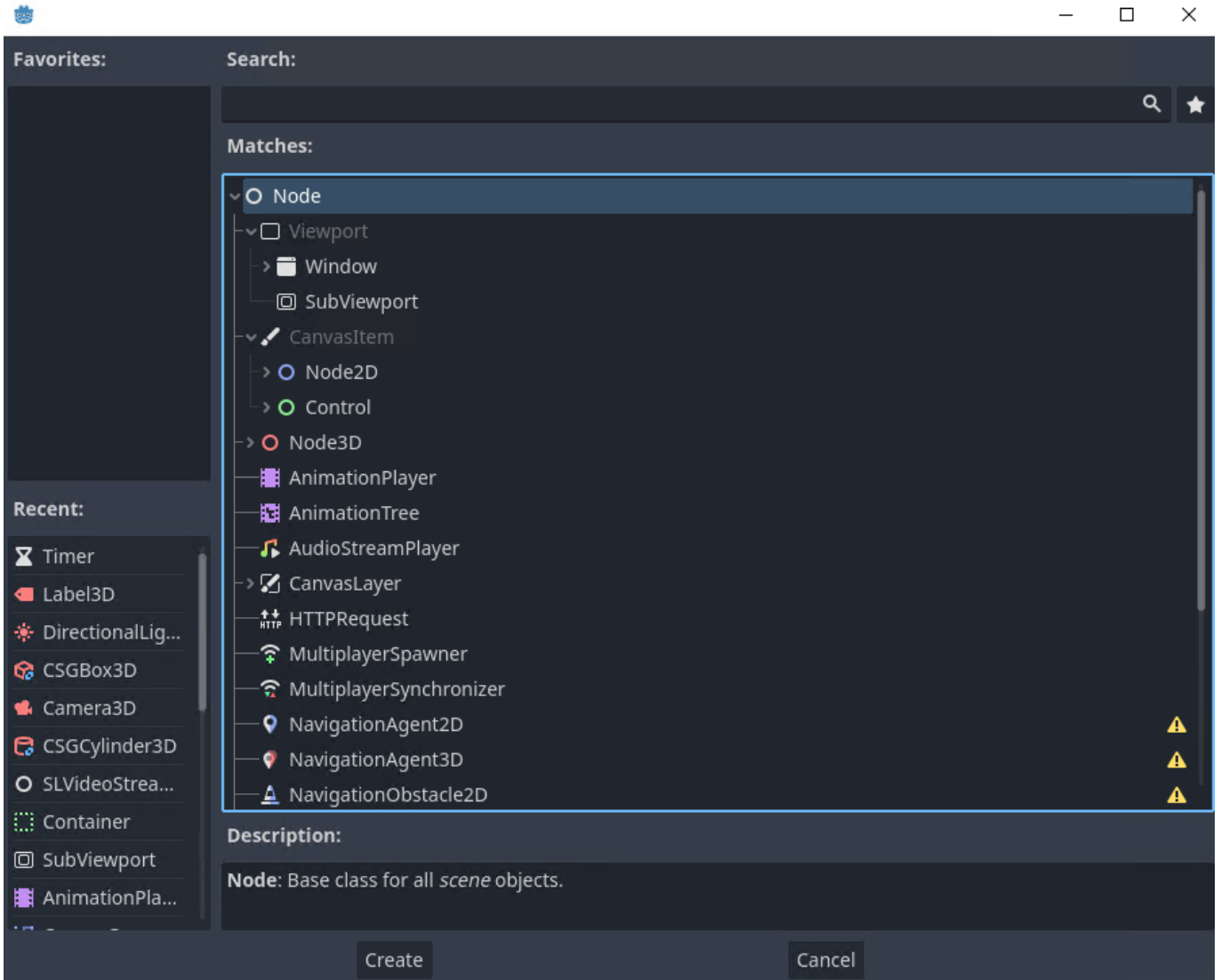
## Skylark Special Objects

The project editor that comes with the Skylark SL NEO distribution currently uses three custom nodes for integration with Skylark.



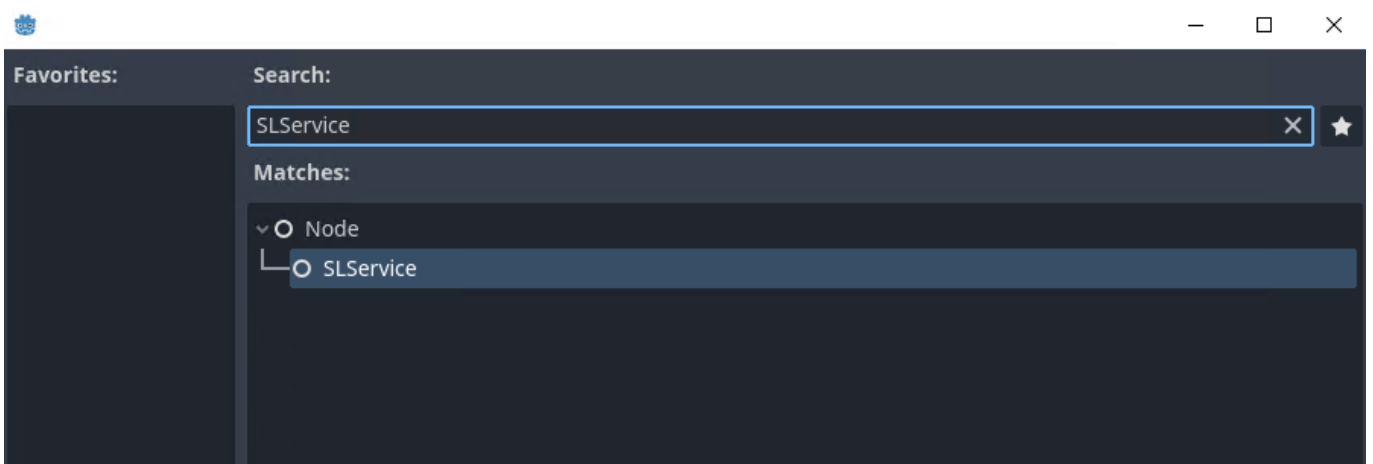


Use the "+" button to add sub nodes to your project's tree.



Select the required node and press Create to add it to your project's tree.

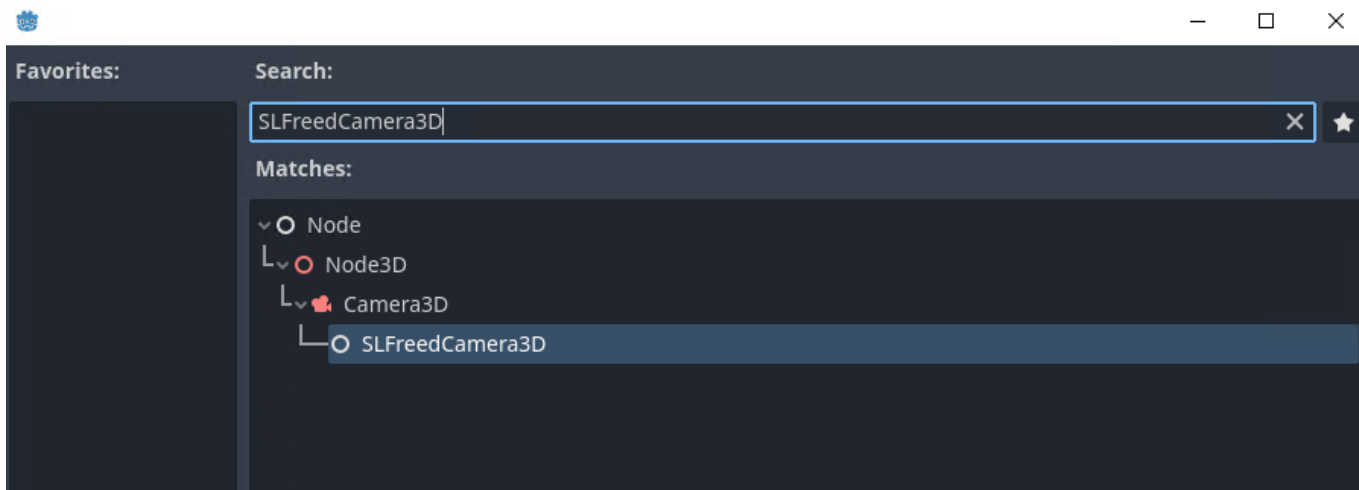
## SLService Node



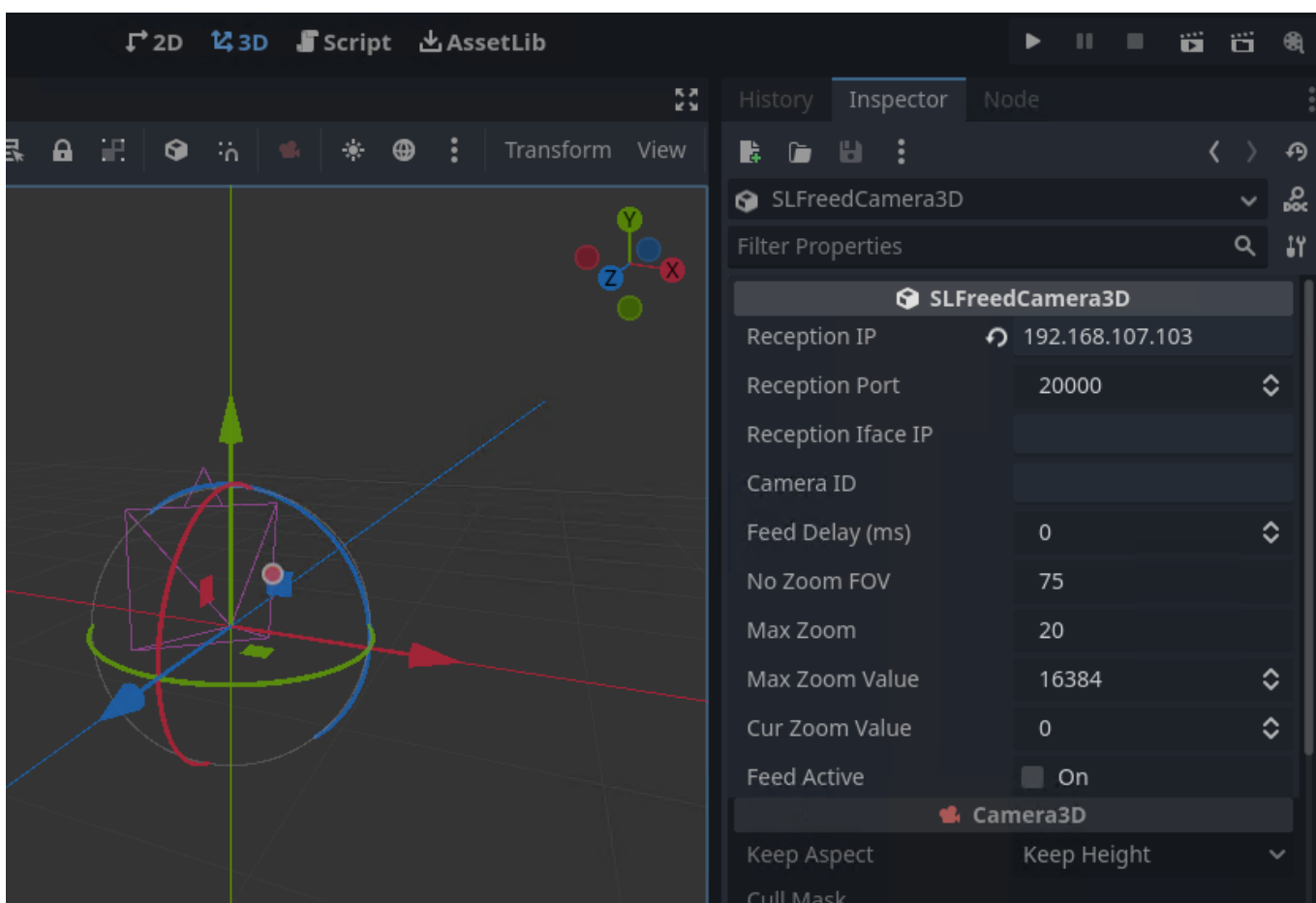
The Node→SLService node is used to exchange commands with Skylark SL Neo [software modules](#):

- receiving commands in the scene is implemented by calling [Godot Action](#).
- passing commands from the scene is done by [command generation](#).

## SLFreedCamera3D Node



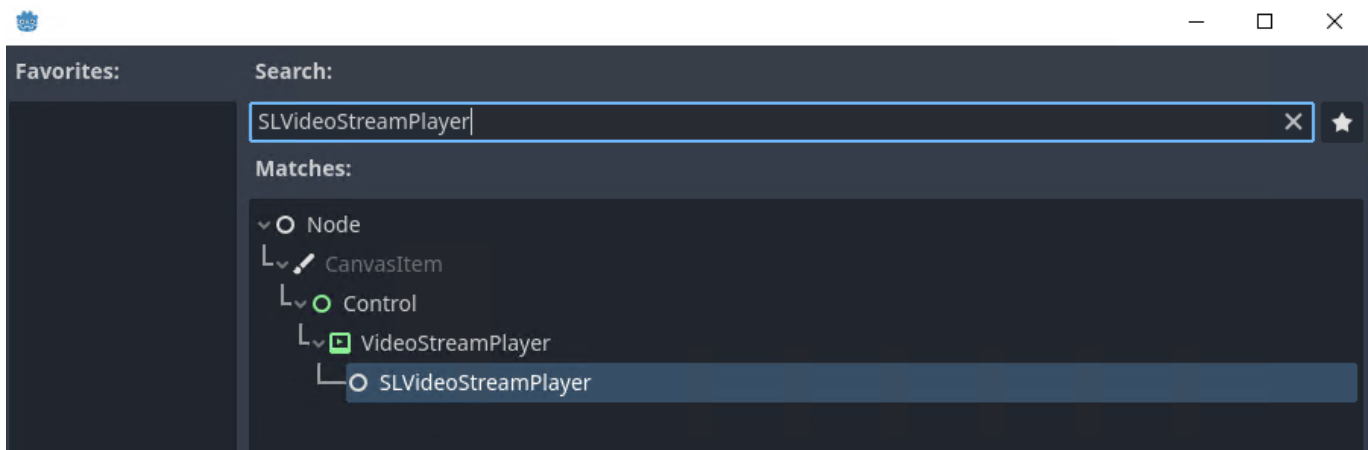
The Node→Node3D→Camera3D→SLFreedCamera3D node allows retrieving the camera movement and zooming data by the Free-D protocol. The Free-D protocol is configured on the camera or tracking device. UDP packets with Free-D data are sent to the address specified in SLFreedCamera3D. You can use the [tracking data simulator](#) to debug projects if a real camera with tracking data is not available.



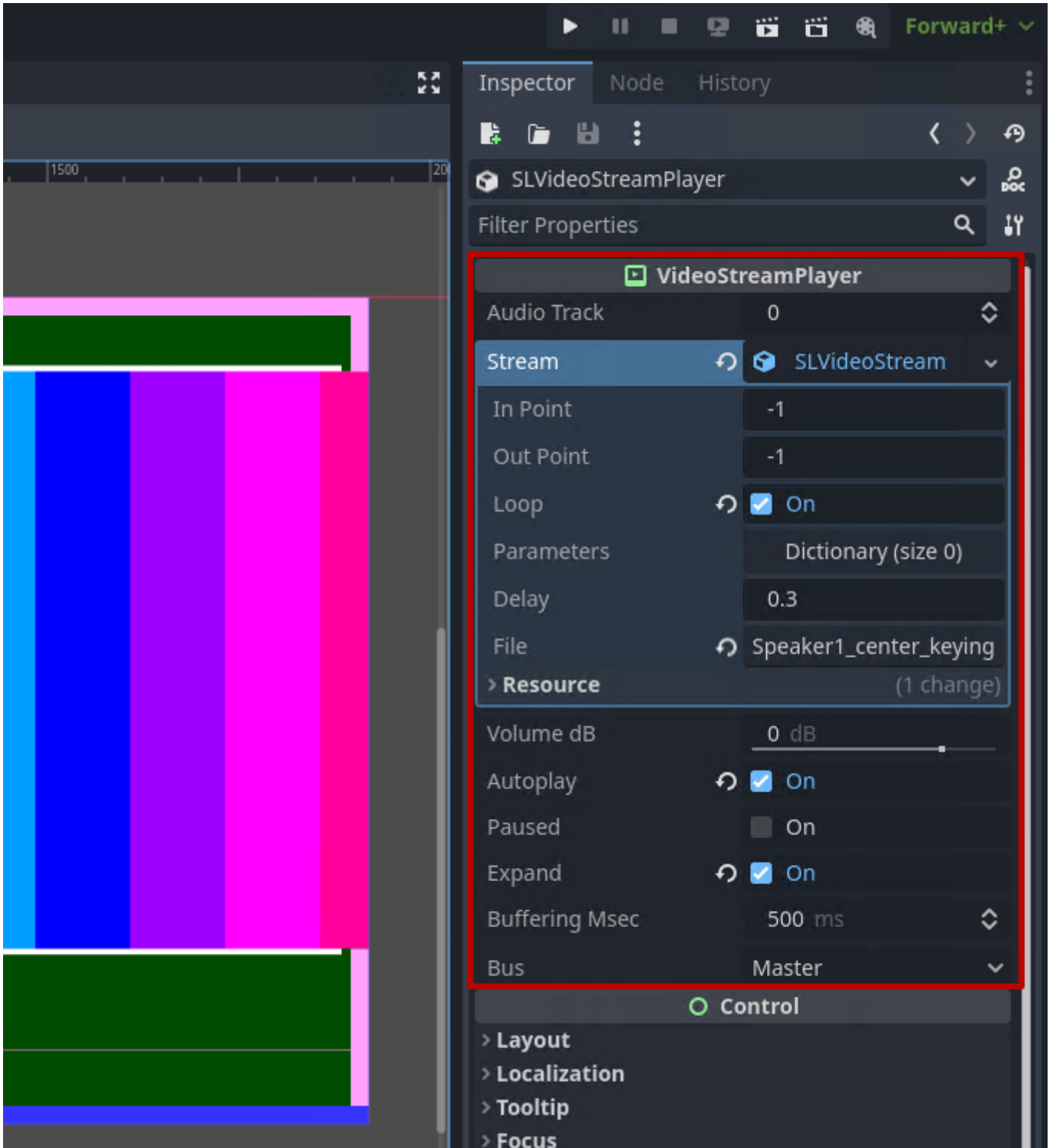
Parameter	Description
Reception IP	IP address used for incoming Free-D protocol connections from the tracking module.
Reception Port	Port for incoming connections from the tracking module.

Parameter	Description
<b>Reception Iface IP</b>	IP address of the interface used to receive the multicast data stream. Used when the Reception IP parameter uses a broadcast address to receive tracking data. The parameter is not filled in when unicast is used.
<b>Camera ID</b>	Fill this parameter if your camera uses a specific CameraID. If the same port and IP address are used to receive Free-D data from different cameras, the filled Camera ID parameter will help to identify the target camera. If the parameter is not specified, tracking data will be received regardless of the received Camera ID value.
<b>Feed Delay (ms)</b>	The parameter defines the delay value of the received tracking data in milliseconds, so that the camera movement corresponds the camera image. Capturing video using Capture→ColorKey→VideoStreamPlayer will lead to video coming with a certain latency due to various buffering. However, the camera position may arrive with much less latency.
<b>No Zoom FOV</b>	Virtual camera view angle corresponding to the camera zero zoom. Specified in degrees.
<b>Max Zoom</b>	Maximum zoom value of the camera in degrees
<b>Max Zoom Value</b>	Maximum zoom value of the camera in conventional units sent in tracking data. Used to convert the retrieved value to degrees. The value may differ depending on the camera manufacturer.
<b>Cur Zoom Value</b>	Current zoom value of the camera in conventional units.
<b>Feed Active</b>	Data receiving activity indicator. Enabled automatically when the camera tracking module connects to the node and sends data.

## SLVideoStreamPlayer Node



Standard node to control playback of video streams:  
Node→CanvasItem→Control→VideoStreamPlayer→SLVideoStreamPlayer. The node supports playing clips from the Skylark SL NEO media database.



Parameter	Description
<b>In Point</b>	In point in seconds. Using fractional values is possible. A dot is used as a delimiter. The default value is -1, in this case the IN point of the played clip will be used.
<b>Out Point</b>	Out point in seconds. Using fractional values is possible. A dot is used as a delimiter. The default value is -1, in this case the OUT point of the played clip will be used.
<b>Loop</b>	If enabled, the clip will be played cyclically, based on the set In/Out Point values.
<b>Parameters</b>	Setting key-value pairs for manual parameterization of uploaded <a href="#">graphic compositions</a> .
<b>Delay</b>	Playback start delay required for preparatory operations (file reading, decoding, buffering, etc.). When designing the logic of scene operation and control commands, consider the clip starting its playback not before the specified time value.

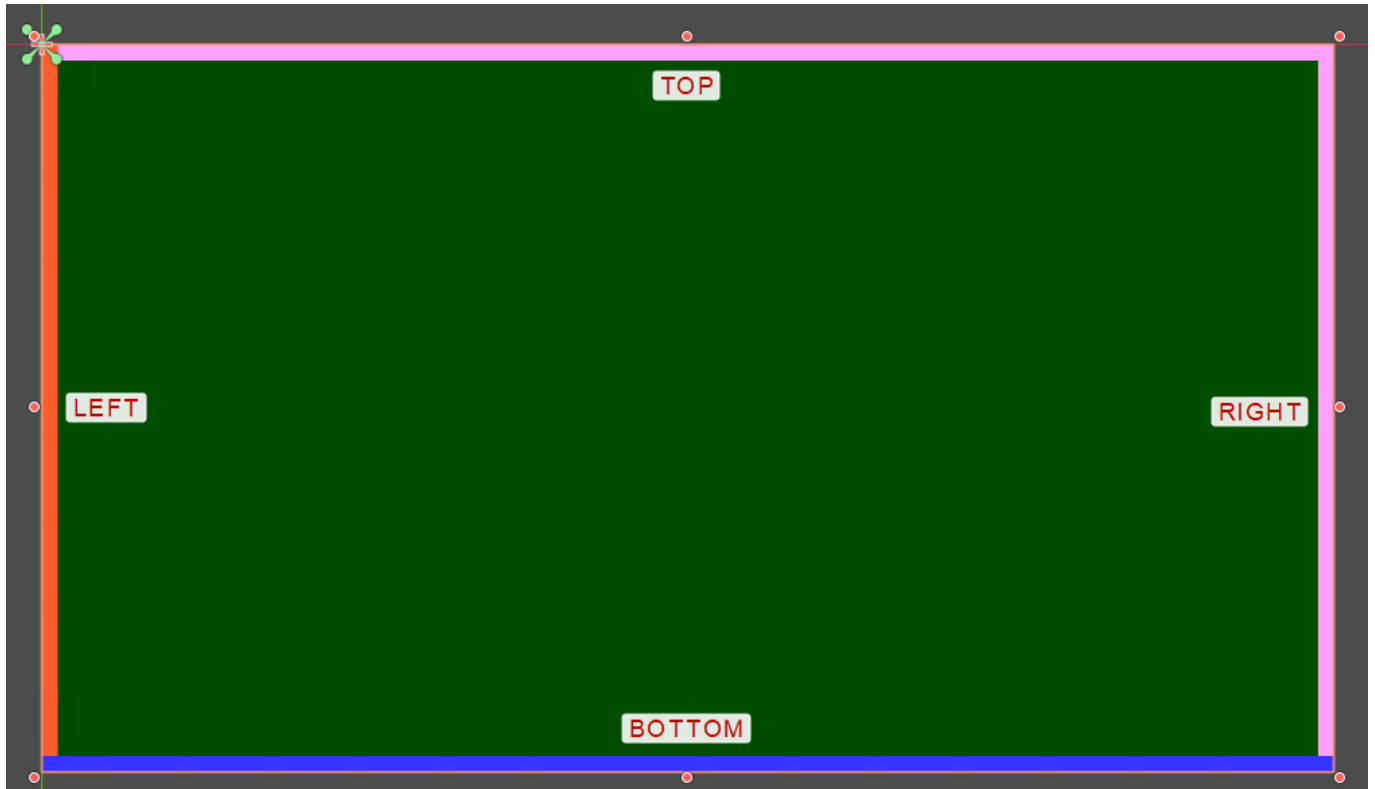
Parameter	Description
<b>File</b>	Name of the played file (video file, graphic composition, live clip, etc.). Possible options: <ul style="list-style-type: none"> <li>• filename - if only the clip name is specified, playback defaults to the first local Skylark SL NEO server database.</li> <li>• medb://192.168.1.10:9902/title:filename - full path to the clip in the media database with the server address and media database port, if the clip is not in the first base.</li> <li>• res://filename - file addressing relative to the project root. These paths work only when running unpacked projects, as reading from ZIP archives is not supported. They are convenient to use on the debugging stage of the project to exclude configuring and launching the media database. Added in version 2.9.123.99.</li> <li>• user://filename - file addressing relative to the user's directory. In Windows, this data is normally located in the directory: %APPDATA%\Godot\app_userdata\<project_name module_name&gt;. 2.9.123.99.<="" added="" in="" li="" version=""> </project_name module_name&gt;.></li></ul>
<b>Local to Scene</b>	Duplicating resources while the scene is loading. Resources are distributed between all instances of the scene by default. Yet there may be situations when a resource property is to be changed at runtime for one object only.
<b>Path</b>	System field
<b>Name</b>	System field
<b>Volume dB</b>	Volume control for the played clip. Possible values are from -80 dB to +24 dB. The 0dB value is used by default - no change to the volume level. If the volume level needs to be decreased, you can set the -80 dB value.
<b>Autoplay</b>	Autoplays the selected clip when the scene starts.
<b>Paused</b>	Pauses playback of the uploaded clip. Pause playback in hidden nodes to optimize resource usage. Resources are used in all cases of video playback, even if the object is not visible on screen.
<b>Expand</b>	Resizing the object.
<b>Buffering Msec</b>	System field
<b>Bus</b>	System field

### Space Orientation

Work with the SLVideoStreamPlayer node may lead to situations when the object is inverted in space: for example, when setting negative values for size or position, or using rotation. Such inversion will make the played video change its orientation in space as well.

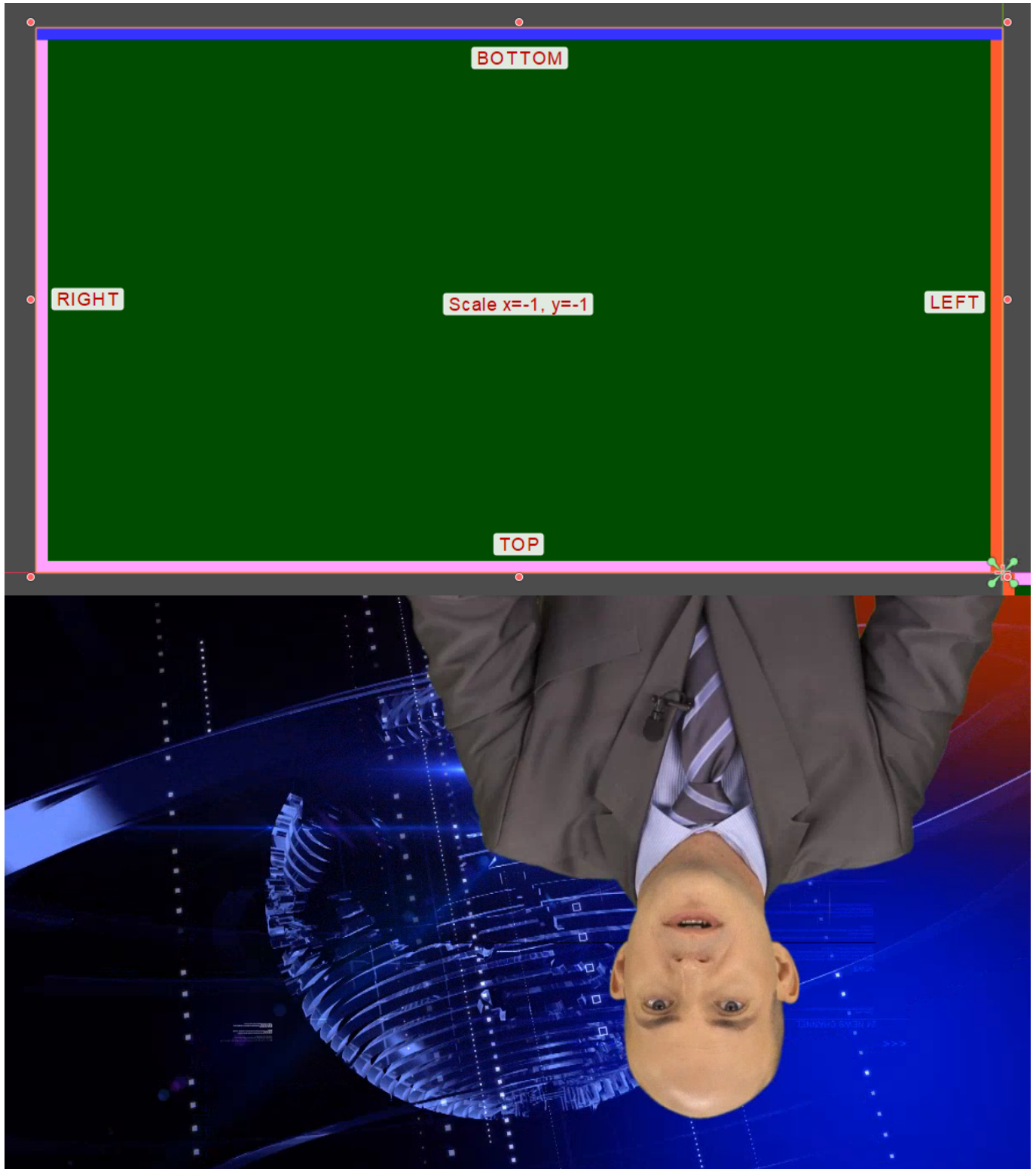
The SLVideoStreamPlayer node has a shader applied to highlight the object sides according to the provided image, so that the correct space orientation of the output video can always be determined.

Below is an example of color coding of the sides corresponding to the default orientation.



Below is an example of the situation when the node has negative Scale values specified.





From:  
<http://wiki.skylark.tv/> - **wiki.skylark.tv**

Permanent link:  
[http://wiki.skylark.tv/applications/client/godot\\_engine](http://wiki.skylark.tv/applications/client/godot_engine)

Last update: **2023/09/05 09:25**

