

SpaceSequenceEditor(Space)

base classes — `bpy_struct`, `Space`

class `bpy.types.SpaceSequenceEditor(Space)`

Sequence editor space data

cache_overlay

Settings for display of overlays

TYPE:

`SequencerCacheOverlay`, (readonly, never None)

cursor_location

2D cursor location for this view

TYPE:

`mathutils.Vector` of 2 items in $[-\text{inf}, \text{inf}]$, default (0.0, 0.0)

display_channel

The channel number shown in the image preview. 0 is the result of all strips combined

TYPE:

int in $[-5, 128]$, default 0

display_mode

View mode to use for displaying sequencer output

TYPE:

enum in ['IMAGE', 'WAVEFORM', 'RGB_PARADE', 'VECTOR_SCOPE', 'HISTOGRAM'], default 'IMAGE'

grease_pencil

Grease Pencil data for this Preview region

TYPE:

`GreasePencil`

overlay_frame_type

Overlay display method

- `RECTANGLE` Rectangle – Show rectangle area overlay.
- `REFERENCE` Reference – Show reference frame only.
- `CURRENT` Current – Show current frame only.

TYPE:

enum in ['RECTANGLE', 'REFERENCE', 'CURRENT'], default 'RECTANGLE'

preview_channels

Channels of the preview to display

- `COLOR_ALPHA` Color & Alpha – Display image with RGB colors and alpha transparency.
- `COLOR` Color – Display image with RGB colors.

TYPE:

enum in ['COLOR_ALPHA', 'COLOR'], default 'COLOR'

. . .

preview_overlay

Settings for display of overlays

TYPE:

`SequencerPreviewOverlay`, (readonly, never None)

proxy_render_size

Display preview using full resolution or different proxy resolutions

TYPE:

enum in ['NONE', 'SCENE', 'PROXY_25', 'PROXY_50', 'PROXY_75', 'PROXY_100'], default 'SCENE'

show_backdrop

Display result under strips

TYPE:

boolean, default False

show_frames

Display frames rather than seconds

TYPE:

boolean, default False

show_gizmo

Show gizmos of all types

TYPE:

boolean, default False

show_gizmo_context

Context sensitive gizmos for the active item

TYPE:

boolean, default False

show_gizmo_navigate

Viewport navigation gizmo

TYPE:

boolean, default False

show_gizmo_tool

Active tool gizmo

TYPE:

boolean, default False

show_markers

If any exists, show markers in a separate row at the bottom of the editor

TYPE:

boolean, default False

show_overexposed

Show overexposed areas with zebra stripes

TYPE:

boolean, default False

mt in [0, 110], default 0

show_overlays

TYPE:

boolean, default False

show_region_channels

TYPE:

boolean, default False

show_region_hud

TYPE:

boolean, default False

show_region_tool_header

TYPE:

boolean, default False

show_region_toolbar

TYPE:

boolean, default False

show_region_ui

TYPE:

boolean, default False

show_seconds

Show timing as a timecode instead of frames

TYPE:

boolean, default False

show_transform_preview

Show a preview of the start or end frame of a strip while transforming its respective handle

TYPE:

boolean, default False

timeline_overlay

Settings for display of overlays

TYPE:

[SequencerTimelineOverlay](#), (readonly, never None)

use_clamp_view

Limit timeline height to maximum used channel slot

TYPE:

boolean, default False

use_marker_sync

Transform markers as well as strips

TYPE:

boolean, default False

use_proxies

Use optimized files for faster scrubbing when available

TYPE:

boolean, default False

use_zoom_to_fit

Automatically zoom preview image to make it fully fit the region

TYPE:

boolean, default False

view_type

Type of the Sequencer view (sequencer, preview or both)

TYPE:

enum in [Space Sequencer View Type Items](#), default 'SEQUENCER'

zoom_percentage

Zoom percentage

TYPE:

float in [0.4, 80000], default 100.0

classmethod bl_ma_get_subclass(id, default=None)

PARAMETERS:

id (*str*) – The RNA type identifier.

RETURNS:

The RNA type or default when not found.

RETURN TYPE:

[bpy.types.Struct](#) subclass

classmethod bl_ma_get_subclass_py(id, default=None)

PARAMETERS:

id (*str*) – The RNA type identifier.

RETURNS:

The class or default when not found.

RETURN TYPE:

type

classmethod draw_handler_add(callback, args, region_type, draw_type)

Add a new draw handler to this space type. It will be called every time the specified region in the space type will be drawn. Note: All arguments are positional only for now.

PARAMETERS:

- **callback** (*Callable[[Any, ...], Any]*) – A function that will be called when the region is drawn. It gets the specified arguments as input, it's return value is ignored.
- **args** (*tuple[Any, ...]*) – Arguments that will be passed to the callback.
- **region_type** (*str*) – The region type the callback draws in; usually `WINDOW`. ([bpy.types.Region.type](#))
- **draw_type** (*str*) – Usually `POST_PIXEL` for 2D drawing and `POST_VIEW` for 3D drawing. In some cases `PRE_VIEW` can be used. `BACKDROP` can be used for backdrops in the node editor.

RETURNS:

Handler that can be removed later on.

RETURN TYPE:

object

classmethod `draw_handler_remove(handler, region_type)`

Remove a draw handler that was added previously.

PARAMETERS:

- **handler** (*object*) – The draw handler that should be removed.
- **region_type** (*str*) – Region type the callback was added to.

Inherited Properties

- `bpy_struct.id_data`
- `Space.show_locked_time`
- `Space.type`
- `Space.show_region_header`

Inherited Functions

- `bpy_struct.as_pointer`
- `bpy_struct.driver_add`
- `bpy_struct.driver_remove`
- `bpy_struct.get`
- `bpy_struct.id_properties_clear`
- `bpy_struct.id_properties_ensure`
- `bpy_struct.id_properties_ui`
- `bpy_struct.is_property_hidden`
- `bpy_struct.is_property_overridable_library`
- `bpy_struct.is_property_readonly`
- `bpy_struct.is_property_set`
- `bpy_struct.items`
- `bpy_struct.keyframe_delete`
- `bpy_struct.keyframe_insert`
- `bpy_struct.keys`
- `bpy_struct.path_from_id`
- `bpy_struct.path_resolve`
- `bpy_struct.pop`
- `bpy_struct.property_overridable_library_set`
- `bpy_struct.property_unset`
- `bpy_struct.type_recast`
- `bpy_struct.values`
- `Space.bl_rna_get_subclass`
- `Space.bl_rna_get_subclass_py`
- `Space.draw_handler_add`
- `Space.draw_handler_remove`