

[Skip to content](#)

TransformConstraint(Constraint)

base classes — [bpy_struct](#), [Constraint](#)

class bpy.types.TransformConstraint(Constraint)

Map transformations of the target to the object

from_max_x

Top range of X axis source motion

TYPE:

float in $[-\text{inf}, \text{inf}]$, default 0.0

from_max_x_rot

Top range of X axis source motion

TYPE:

float in $[-\text{inf}, \text{inf}]$, default 0.0

from_max_x_scale

Top range of X axis source motion

TYPE:

float in $[-\text{inf}, \text{inf}]$, default 0.0

from_max_y

Top range of Y axis source motion

TYPE:

float in $[-\text{inf}, \text{inf}]$, default 0.0

from_max_y_rot

Top range of Y axis source motion

TYPE:

float in $[-\text{inf}, \text{inf}]$, default 0.0

from_max_y_scale

Top range of Y axis source motion

TYPE:

float in $[-\text{inf}, \text{inf}]$, default 0.0

from_max_z

Top range of Z axis source motion

TYPE:

float in $[-\text{inf}, \text{inf}]$, default 0.0

from_max_z_rot

Top range of Z axis source motion

TYPE:

float in $[-\text{inf}, \text{inf}]$, default 0.0

from_max_z_scale

Top range of Z axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_min_x

Bottom range of X axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_min_x_rot

Bottom range of X axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_min_x_scale

Bottom range of X axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_min_y

Bottom range of Y axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_min_y_rot

Bottom range of Y axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_min_y_scale

Bottom range of Y axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_min_z

Bottom range of Z axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_min_z_rot

Bottom range of Z axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_min_z_scale

Bottom range of Z axis source motion

TYPE:

float in [-inf, inf], default 0.0

from_rotation_mode

Specify the type of rotation channels to use

TYPE:

enum in [Driver Target Rotation Mode Items](#), default 'AUTO'

map_from

The transformation type to use from the target

TYPE:

enum in ['LOCATION', 'ROTATION', 'SCALE'], default 'LOCATION'

map_to

The transformation type to affect on the constrained object

TYPE:

enum in ['LOCATION', 'ROTATION', 'SCALE'], default 'LOCATION'

map_to_x_from

The source axis constrained object's X axis uses

TYPE:

enum in [Axis Xyz Items](#), default 'X'

map_to_y_from

The source axis constrained object's Y axis uses

TYPE:

enum in [Axis Xyz Items](#), default 'X'

map_to_z_from

The source axis constrained object's Z axis uses

TYPE:

enum in [Axis Xyz Items](#), default 'X'

mix_mode

Specify how to combine the new location with original

- `REPLACE` Replace – Replace component values.
- `ADD` Add – Add component values together.

TYPE:

enum in ['REPLACE', 'ADD'], default 'ADD'

mix_mode_rot

Specify how to combine the new rotation with original

- `REPLACE` Replace – Replace component values.
- `ADD` Add – Add component values together.
- `BEFORE` Before Original – Apply new rotation before original, as if it was on a parent.
- `AFTER` After Original – Apply new rotation after original, as if it was on a child.

TYPE:

enum in ['REPLACE', 'ADD', 'BEFORE', 'AFTER'], default 'ADD'

mix_mode_scale

Specify how to combine the new scale with original

- `REPLACE` Replace – Replace component values.
- `MULTIPLY` Multiply – Multiply component values together.

TYPE:

enum in ['REPLACE', 'MULTIPLY'], default 'REPLACE'

subtarget

Armature bone, mesh or lattice vertex group, ...

TYPE:

string, default "", (never None)

target

Target object

TYPE:

[Object](#)

to_euler_order

Explicitly specify the output euler rotation order

- `AUTO` Default – Euler using the default rotation order.
- `XYZ` XYZ Euler – Euler using the XYZ rotation order.
- `XZY` XZY Euler – Euler using the XZY rotation order.
- `YXZ` YXZ Euler – Euler using the YXZ rotation order.
- `YZX` YZX Euler – Euler using the YZX rotation order.
- `ZXY` ZXY Euler – Euler using the ZXY rotation order.
- `ZYX` ZYX Euler – Euler using the ZYX rotation order.

TYPE:

enum in ['AUTO', 'XYZ', 'XZY', 'YXZ', 'YZX', 'ZXY', 'ZYX'], default 'AUTO'

to_max_x

Top range of X axis destination motion

TYPE:

float in [-inf, inf], default 0.0

to_max_x_rot

Top range of X axis destination motion

TYPE:

float in [-inf, inf], default 0.0

to_max_x_scale

Top range of X axis destination motion

TYPE:

float in [-inf, inf], default 0.0

to_max_y

Top range of Y axis destination motion

TYPE:

float in [-inf, inf], default 0.0

to_max_y_rot

Top range of Y axis destination motion

TYPE:

float in $[-\infty, \infty]$, default 0.0

to_max_y_scale

Top range of Y axis destination motion

TYPE:

float in $[-\infty, \infty]$, default 0.0

to_max_z

Top range of Z axis destination motion

TYPE:

float in $[-\infty, \infty]$, default 0.0

to_max_z_rot

Top range of Z axis destination motion

TYPE:

float in $[-\infty, \infty]$, default 0.0

to_max_z_scale

Top range of Z axis destination motion

TYPE:

float in $[-\infty, \infty]$, default 0.0

to_min_x

Bottom range of X axis destination motion

TYPE:

float in $[-\infty, \infty]$, default 0.0

to_min_x_rot

Bottom range of X axis destination motion

TYPE:

float in $[-\infty, \infty]$, default 0.0

to_min_x_scale

Bottom range of X axis destination motion

TYPE:

float in $[-\infty, \infty]$, default 0.0

to_min_y

Bottom range of Y axis destination motion

TYPE:

float in $[-\infty, \infty]$, default 0.0

to_min_y_rot

Bottom range of Y axis destination motion

TYPE:

.....

float in [-inf, inf], default 0.0

to_min_y_scale

Bottom range of Y axis destination motion

TYPE:

float in [-inf, inf], default 0.0

to_min_z

Bottom range of Z axis destination motion

TYPE:

float in [-inf, inf], default 0.0

to_min_z_rot

Bottom range of Z axis destination motion

TYPE:

float in [-inf, inf], default 0.0

to_min_z_scale

Bottom range of Z axis destination motion

TYPE:

float in [-inf, inf], default 0.0

use_motion_extrapolate

Extrapolate ranges

TYPE:

boolean, default False

classmethod bl_rna_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_rna_get_subclass_py(id, default=None)

PARAMETERS:

id (*str*) – The RNA type identifier.

RETURNS:

The class or default when not found.

RETURN TYPE:

type

Inherited Properties

- `bpy_struct.id_data`
- `Constraint.name`
- `Constraint.type`
- `Constraint.mute`
- `Constraint.enabled`
- `Constraint.show_expanded`

- `Constraint.is_override_data`
- `Constraint.is_valid`
- `Constraint.owner_space`
- `Constraint.active`
- `Constraint.target_space`
- `Constraint.influence`
- `Constraint.space_object`
- `Constraint.error_location`
- `Constraint.space_subtarget`
- `Constraint.error_rotation`

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`
- `Constraint.bl_rna_get_subclass`
- `Constraint.bl_rna_get_subclass_py`