

[Skip to content](#)

# MarbleTexture(Texture)

base classes — [bpy\\_struct](#), [ID](#), [Texture](#)

**class** bpy.types.MarbleTexture(Texture)

Procedural noise texture

## marble\_type

- `SOFT` Soft – Use soft marble.
- `SHARP` Sharp – Use more clearly defined marble.
- `SHARPER` Sharper – Use very clearly defined marble.

### TYPE:

enum in ['SOFT', 'SHARP', 'SHARPER'], default 'SOFT'

## nabla

Size of derivative offset used for calculating normal

### TYPE:

float in [0.001, 0.1], default 0.025

## noise\_basis

Noise basis used for turbulence

- `BLENDER_ORIGINAL` Blender Original – Noise algorithm - Blender original: Smooth interpolated noise.
- `ORIGINAL_PERLIN` Original Perlin – Noise algorithm - Original Perlin: Smooth interpolated noise.
- `IMPROVED_PERLIN` Improved Perlin – Noise algorithm - Improved Perlin: Smooth interpolated noise.
- `VORONOI_F1` Voronoi F1 – Noise algorithm - Voronoi F1: Returns distance to the closest feature point.
- `VORONOI_F2` Voronoi F2 – Noise algorithm - Voronoi F2: Returns distance to the 2nd closest feature point.
- `VORONOI_F3` Voronoi F3 – Noise algorithm - Voronoi F3: Returns distance to the 3rd closest feature point.
- `VORONOI_F4` Voronoi F4 – Noise algorithm - Voronoi F4: Returns distance to the 4th closest feature point.
- `VORONOI_F2_F1` Voronoi F2-F1 – Noise algorithm - Voronoi F1-F2.
- `VORONOI_CRACKLE` Voronoi Crackle – Noise algorithm - Voronoi Crackle: Voronoi tessellation with sharp edges.
- `CELL_NOISE` Cell Noise – Noise algorithm - Cell Noise: Square cell tessellation.

### TYPE:

enum in ['BLENDER\_ORIGINAL', 'ORIGINAL\_PERLIN', 'IMPROVED\_PERLIN', 'VORONOI\_F1', 'VORONOI\_F2', 'VORONOI\_F3', 'VORONOI\_F4', 'VORONOI\_F2\_F1', 'VORONOI\_CRACKLE', 'CELL\_NOISE'], default 'BLENDER\_ORIGINAL'

## noise\_basis\_2

- `SIN` Sin – Use a sine wave to produce bands.
- `SAW` Saw – Use a saw wave to produce bands.
- `TRI` Tri – Use a triangle wave to produce bands.

### TYPE:

enum in ['SIN', 'SAW', 'TRI'], default 'SIN'

## noise\_depth

Depth of the cloud calculation

### TYPE:

int in [0, 30], default 2

### noise\_scale

Scaling for noise input

#### TYPE:

float in [0.0001, inf], default 0.25

### noise\_type

- `SOFT_NOISE` Soft – Generate soft noise (smooth transitions).
- `HARD_NOISE` Hard – Generate hard noise (sharp transitions).

#### TYPE:

enum in ['SOFT\_NOISE', 'HARD\_NOISE'], default 'SOFT\_NOISE'

### turbulence

Turbulence of the bandnoise and ringnoise types

#### TYPE:

float in [0.0001, inf], default 5.0

### users\_material

Materials that use this texture

(readonly)

### users\_object\_modifier

Object modifiers that use this texture

(readonly)

### 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

## Inherited Properties

- `bpy_struct.id_data`
- `ID.name`
- `ID.name_full`
- `ID.id_type`
- `ID.session_uid`
- `ID.is_evaluated`
- `ID.override_library`
- `ID.preview`
- `Texture.type`
- `Texture.use_clamp`
- `Texture.use_color_ramp`

- `ID.is_evaluated`
- `ID.original`
- `ID.users`
- `ID.use_fake_user`
- `ID.use_extra_user`
- `ID.is_embedded_data`
- `ID.is_missing`
- `ID.is_runtime_data`
- `ID.is_editable`
- `ID.tag`
- `ID.is_library_indirect`
- `ID.library`
- `ID.library_weak_reference`
- `ID.asset_data`
- `Texture.color_ramp`
- `Texture.intensity`
- `Texture.contrast`
- `Texture.saturation`
- `Texture.factor_red`
- `Texture.factor_green`
- `Texture.factor_blue`
- `Texture.use_preview_alpha`
- `Texture.use_nodes`
- `Texture.node_tree`
- `Texture.animation_data`
- `Texture.users_material`
- `Texture.users_object_modifier`

## 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`
- `ID.rename`
- `ID.evaluated_get`
- `ID.copy`
- `ID.asset_mark`
- `ID.asset_clear`
- `ID.asset_generate_preview`
- `ID.override_create`
- `ID.override_hierarchy_create`
- `ID.user_clear`
- `ID.user_remap`
- `ID.make_local`
- `ID.user_of_id`
- `ID.animation_data_create`
- `ID.animation_data_clear`
- `ID.update_tag`
- `ID.preview_ensure`
- `ID.bl_rna_get_subclass`
- `ID.bl_rna_get_subclass_py`
- `Texture.evaluate`
- `Texture.bl_rna_get_subclass`
- `Texture.bl_rna_get_subclass_py`