FluidDomainSettings(bpy_struct)

base class — bpy_struct

class bpy.types.**FluidDomainSettings(bpy_struct)**

Fluid domain settings

adapt_margin

Margin added around fluid to minimize boundary interference

TYPE:

int in [2, 24], default 4

adapt_threshold

Minimum amount of fluid a cell can contain before it is considered empty

TYPE:

float in [0, 1], default 0.02

additional_res

Maximum number of additional cells

TYPE:

int in [0, 512], default 0

alpha

Buoyant force based on smoke density (higher value results in faster rising smoke)

TYPE:

float in [-5, 5], default 1.0

beta

Buoyant force based on smoke heat (higher value results in faster rising smoke)

TYPE:

float in [-5, 5], default 1.0

burning_rate

Speed of the burning reaction (higher value results in smaller flames)

TYPE:

float in [0.01, 4], default 0.75

cache_data_format

Select the file format to be used for caching volumetric data

- `UNI` Uni Cache – Uni file format (.uni).
- `OPENVDB` OpenVDB – OpenVDB file format (.vdb).
- `RAW` Raw Cache – Raw file format (.raw).

TYPE:

enum in ['UNI', 'OPENVDB', 'RAW'], default 'OPENVDB'

cache_directory

Directory that contains fluid cache files

— —

TYPE:

string, default ‘’, (never None)

cache_frame_end

Frame on which the simulation stops (last frame baked)

TYPE:

int in [-1048574, 1048574], default 250

cache_frame_offset

Frame offset that is used when loading the simulation from the cache. It is not considered when baking the simulation, only when loading it.

TYPE:

int in [-1048574, 1048574], default 0

cache_frame_pause_data

TYPE:

int in [-inf, inf], default 0

cache_frame_pause_guide

TYPE:

int in [-inf, inf], default 0

cache_frame_pause_mesh

TYPE:

int in [-inf, inf], default 0

cache_frame_pause_noise

TYPE:

int in [-inf, inf], default 0

cache_frame_pause_particles

TYPE:

int in [-inf, inf], default 0

cache_frame_start

Frame on which the simulation starts (first frame baked)

TYPE:

int in [-1048574, 1048574], default 1

cache_mesh_format

Select the file format to be used for caching surface data

- `UNI` Uni Cache – Uni file format (.uni).
- `OPENVDB` OpenVDB – OpenVDB file format (.vdb).
- `RAW` Raw Cache – Raw file format (.raw).

TYPE:

enum in [‘UNI’, ‘OPENVDB’, ‘RAW’], default ‘UNI’

cache_noise_format

Select the file format to be used for caching noise data

- `UNI` Uni Cache – Uni file format (.uni).
- `OPENVDB` OpenVDB – OpenVDB file format (.vdb).

- `RAW` Raw Cache – Raw file format (.raw).

TYPE:

enum in ['UNI', 'OPENVDB', 'RAW'], default 'OPENVDB'

cache_particle_format

Select the file format to be used for caching particle data

- `UNI` Uni Cache – Uni file format (.uni).
- `OPENVDB` OpenVDB – OpenVDB file format (.vdb).
- `RAW` Raw Cache – Raw file format (.raw).

TYPE:

enum in ['UNI', 'OPENVDB', 'RAW'], default 'OPENVDB'

cache_resumable

Additional data will be saved so that the bake jobs can be resumed after pausing. Because more data will be written to disk it is recommended to avoid enabling this option when baking at high resolutions.

TYPE:

boolean, default False

cache_type

Change the cache type of the simulation

- `REPLAY` Replay – Use the timeline to bake the scene.
- `MODULAR` Modular – Bake every stage of the simulation separately.
- `ALL` All – Bake all simulation settings at once.

TYPE:

enum in ['REPLAY', 'MODULAR', 'ALL'], default 'REPLAY'

cell_size

Cell Size

TYPE:

`mathutils.Vector` of 3 items in [-inf, inf], default (0.0, 0.0, 0.0), (readonly)

cfl_condition

Maximal velocity per cell (greater CFL numbers will minimize the number of simulation steps and the computation time.)

TYPE:

float in [0, 10], default 2.0

clipping

Value under which voxels are considered empty space to optimize rendering

TYPE:

float in [0, 1], default 1e-06

color_grid

Smoke color grid

TYPE:

float array of 32 items in [-inf, inf], default (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0), (readonly)

color_ramp

TYPE:

ColorRamp, (readonly)

color_ramp_field

Simulation field to color map

TYPE:

enum in ['NONE'], default 'NONE'

color_ramp_field_scale

Multiplier for scaling the selected field to color map

TYPE:

float in [0.001, 100000], default 1.0

delete_in_obstacle

Delete fluid inside obstacles

TYPE:

boolean, default False

density_grid

Smoke density grid

TYPE:

float array of 32 items in [-inf, inf], default (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0), (readonly)

display_interpolation

Interpolation method to use for smoke/fire volumes in solid mode

- LINEAR Linear – Good smoothness and speed.
- CUBIC Cubic – Smoothed high quality interpolation, but slower.
- CLOSEST Closest – No interpolation.

TYPE:

enum in ['LINEAR', 'CUBIC', 'CLOSEST'], default 'LINEAR'

display_thickness

Thickness of smoke display in the viewport

TYPE:

float in [0.001, 1000], default 1.0

dissolve_speed

Determine how quickly the smoke dissolves (lower value makes smoke disappear faster)

TYPE:

int in [1, 10000], default 5

domain_resolution

Smoke Grid Resolution

TYPE:

int array of 3 items in [-inf, inf], default (0, 0, 0), (readonly)

domain_type

Choose domain type of the simulation

Change domain type of the simulation

- `GAS` Gas – Create domain for gases.
- `LIQUID` Liquid – Create domain for liquids.

TYPE:

enum in ['GAS', 'LIQUID'], default 'GAS'

effector_group

Limit effectors to this collection

TYPE:

`Collection`

effector_weights

TYPE:

`EffectorWeights`, (readonly)

export_manta_script

Generate and export Mantaflow script from current domain settings during bake. This is only needed if you plan to analyze the cache (e.g. vie grids, velocity vectors, particles) in Mantaflow directly (outside of Blender) after baking the simulation.

TYPE:

boolean, default False

flame_grid

Smoke flame grid

TYPE:

float array of 32 items in [-inf, inf], default (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0), (readonly)

flame_ignition

Minimum temperature of the flames (higher value results in faster rising flames)

TYPE:

float in [0.5, 5], default 1.5

flame_max_temp

Maximum temperature of the flames (higher value results in faster rising flames)

TYPE:

float in [1, 10], default 3.0

flame_smoke

Amount of smoke created by burning fuel

TYPE:

float in [0, 8], default 1.0

flame_smoke_color

Color of smoke emitted from burning fuel

TYPE:

`mathutils.Color` of 3 items in [0, inf], default (0.7, 0.7, 0.7)

flame_vorticity

Additional vorticity for the flames

TYPE:

float in [0, 2], default 0.5

flip_ratio

PIC/FLIP Ratio. A value of 1.0 will result in a completely FLIP based simulation. Use a lower value for simulations which should produce smaller splashes.

TYPE:

float in [0, 1], default 0.97

fluid_group

Limit fluid objects to this collection

TYPE:

`Collection`

force_collection

Limit forces to this collection

TYPE:

`Collection`

fractions_distance

Determines how far apart fluid and obstacle are (higher values will result in fluid being further away from obstacles, smaller values will let fluid move towards the inside of obstacles)

TYPE:

float in [-5, 5], default 0.5

fractions_threshold

Determines how much fluid is allowed in an obstacle cell (higher values will tag a boundary cell as an obstacle easier and reduce the boundary smoothening effect)

TYPE:

float in [0.001, 1], default 0.05

gravity

Gravity in X, Y and Z direction

TYPE:

`mathutils.Vector` of 3 items in [-1000.1, 1000.1], default (0.0, 0.0, -9.81)

gridlines_cell_filter

Cell type to be highlighted

- `NONE` None – Highlight the cells regardless of their type.
- `FLUID` Fluid – Highlight only the cells of type Fluid.
- `OBSTACLE` Obstacle – Highlight only the cells of type Obstacle.
- `EMPTY` Empty – Highlight only the cells of type Empty.
- `INFLOW` Inflow – Highlight only the cells of type Inflow.
- `OUTFLOW` Outflow – Highlight only the cells of type Outflow.

TYPE:

enum in ['NONE', 'FLUID', 'OBSTACLE', 'EMPTY', 'INFLOW', 'OUTFLOW'], default 'NONE'

gridlines_color_field

Simulation field to color map onto gridlines

- `NONE` None – None.
- `FLAGS` Flags – Flag grid of the fluid domain.
- `RANGE` Highlight Range – Highlight the voxels with values of the color mapped field within the range.

TYPE:

enum in ['NONE', 'FLAGS', 'RANGE'], default 'NONE'

gridlines_lower_bound

Lower bound of the highlighting range

TYPE:

float in [-inf, inf], default 0.0

gridlines_range_color

Color used to highlight the range

TYPE:

float array of 4 items in [0, inf], default (1.0, 0.0, 0.0, 1.0)

gridlines_upper_bound

Upper bound of the highlighting range

TYPE:

float in [-inf, inf], default 1.0

guide_alpha

Guiding weight (higher value results in greater lag)

TYPE:

float in [1, 100], default 2.0

guide_beta

Guiding size (higher value results in larger vortices)

TYPE:

int in [1, 50], default 5

guide_parent

Use velocities from this object for the guiding effect (object needs to have fluid modifier and be of type domain))

TYPE:

`Object`

guide_source

Choose where to get guiding velocities from

- `DOMAIN` Domain – Use a fluid domain for guiding (domain needs to be baked already so that velocities can be extracted). Guiding dom can be of any type (i.e. gas or liquid)..
- `EFFECTOR` Effector – Use guiding (effector) objects to create fluid guiding (guiding objects should be animated and baked once set up completely).

TYPE:

enum in ['DOMAIN', 'EFFECTOR'], default 'DOMAIN'

guide_vel_factor

Guiding velocity factor (higher value results in greater guiding velocities)

TYPE:

float in [0, 100], default 2.0

has_cache_baked_any

TYPE:

boolean, default False

has_cache_baked_data

TYPE:

boolean, default False

has_cache_baked_guide

TYPE:

boolean, default False

has_cache_baked_mesh

TYPE:

boolean, default False

has_cache_baked_noise

TYPE:

boolean, default False

has_cache_baked_particles

TYPE:

boolean, default False

heat_grid

Smoke heat grid

TYPE:

float array of 32 items in [-inf, inf], default (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0), (readonly)

highres_sampling

Method for sampling the high resolution flow

TYPE:

enum in ['FULLSAMPLE', 'LINEAR', 'NEAREST'], default 'FULLSAMPLE'

is_cache_baking_any

TYPE:

boolean, default False

is_cache_baking_data

TYPE:

boolean, default False

is_cache_baking_guide

TYPE:

boolean, default False

is_cache_baking_mesh

TYPE:

boolean, default False

is_cache_baking_noise

TYPE:

boolean, default False

is_cache_baking_particles

TYPE:

boolean, default False

mesh_concave_lower

Lower mesh concavity bound (high values tend to smoothen and fill out concave regions)

TYPE:

float in [0, 10], default 0.4

mesh_concave_upper

Upper mesh concavity bound (high values tend to smoothen and fill out concave regions)

TYPE:

float in [0, 10], default 3.5

mesh_generator

Which particle level set generator to use

- `IMPROVED` Final – Use improved particle level set (slower but more precise and with mesh smoothening options).
- `UNION` Preview – Use union particle level set (faster but lower quality).

TYPE:

enum in ['IMPROVED', 'UNION'], default 'IMPROVED'

mesh_particle_radius

Particle radius factor (higher value results in larger (meshed) particles). Needs to be adjusted after changing the mesh scale.

TYPE:

float in [0, 10], default 2.0

mesh_scale

The mesh simulation is scaled up by this factor (compared to the base resolution of the domain). For best meshing, it is recommended to adjus the mesh particle radius alongside this value.

TYPE:

int in [1, 100], default 2

mesh_smoothen_neg

Negative mesh smoothening

TYPE:

int in [0, 100], default 1

mesh_smoothen_pos

Positive mesh smoothening

TYPE:

int in [0, 100], default 1

noise_pos_scale

Scale of noise (higher value results in larger vortices)

TYPE:

float in [0.0001, 10], default 2.0

noise_scale

The noise simulation is scaled up by this factor (compared to the base resolution of the domain)

TYPE:

int in [1, 100], default 2

noise_strength

Strength of noise

TYPE:

float in [0, 10], default 1.0

noise_time_anim

Animation time of noise

TYPE:

float in [0.0001, 10], default 0.1

openvdb_cache_compress_type

Compression method to be used

- `ZIP` Zip – Effective but slow compression.
- `BLOSC` Blosc – Multithreaded compression, similar in size and quality as 'Zip'.
- `NONE` None – Do not use any compression.

TYPE:

enum in ['ZIP', 'BLOSC', 'NONE'], default 'BLOSC'

openvdb_data_depth

Bit depth for fluid particles and grids (lower bit values reduce file size)

TYPE:

enum in ['NONE'], default 'NONE'

particle_band_width

Particle (narrow) band width (higher value results in thicker band and more particles)

TYPE:

float in [0, 1000], default 3.0

particle_max

Maximum number of particles per cell (ensures that each cell has at most this amount of particles)

TYPE:

int in [0, 1000], default 16

particle_min

Minimum number of particles per cell (ensures that each cell has at least this amount of particles)

TYPE:

int in [0, 1000], default 8

particle_number

Particle number factor (higher value results in more particles)

TYPE:

int in [1, 5], default 2

particle_radius

Particle radius factor. Increase this value if the simulation appears to leak volume, decrease it if the simulation seems to gain volume.

TYPE:

float in [0, 10], default 1.0

particle_randomness

Randomness factor for particle sampling

TYPE:

float in [0, 10], default 0.1

particle_scale

The particle simulation is scaled up by this factor (compared to the base resolution of the domain)

TYPE:

int in [1, 100], default 1

resolution_max

Resolution used for the fluid domain. Value corresponds to the longest domain side (resolution for other domain sides is calculated automatically).

TYPE:

int in [6, 10000], default 32

show_gridlines

Show gridlines

TYPE:

boolean, default False

show_velocity

Visualize vector fields

TYPE:

boolean, default False

simulation_method

Change the underlying simulation method

- `FLIP` FLIP – Use FLIP as the simulation method (more splashy behavior).
- `APIC` APIC – Use APIC as the simulation method (more energetic and stable behavior).

TYPE:

enum in ['FLIP', 'APIC'], default 'FLIP'

slice_axis

- `AUTO` Auto – Adjust slice direction according to the view direction.
- `X` X – Slice along the X axis.
- `Y` Y – Slice along the Y axis.
- `Z` Z – Slice along the Z axis.

TYPE:

enum in [‘AUTO’, ‘X’, ‘Y’, ‘Z’], default ‘AUTO’

slice_depth

Position of the slice

TYPE:

float in [0, 1], default 0.5

slice_per_voxel

How many slices per voxel should be generated

TYPE:

float in [0, 100], default 5.0

sndparticle_boundary

How particles that left the domain are treated

- `DELETE` Delete – Delete secondary particles that are inside obstacles or left the domain.
- `PUSHOUT` Push Out – Push secondary particles that left the domain back into the domain.

TYPE:

enum in [‘DELETE’, ‘PUSHOUT’], default ‘DELETE’

sndparticle_bubble_buoyancy

Amount of buoyancy force that rises bubbles (high value results in bubble movement mainly upwards)

TYPE:

float in [0, 100], default 0.5

sndparticle_bubble_drag

Amount of drag force that moves bubbles along with the fluid (high value results in bubble movement mainly along with the fluid)

TYPE:

float in [0, 100], default 0.6

sndparticle_combined_export

Determines which particle systems are created from secondary particles

- `OFF` Off – Create a separate particle system for every secondary particle type.
- `SPRAY_FOAM` Spray + Foam – Spray and foam particles are saved in the same particle system.
- `SPRAY_BUBBLES` Spray + Bubbles – Spray and bubble particles are saved in the same particle system.
- `FOAM_BUBBLES` Foam + Bubbles – Foam and bubbles particles are saved in the same particle system.
- `SPRAY_FOAM_BUBBLES` Spray + Foam + Bubbles – Create one particle system that contains all three secondary particle types.

TYPE:

enum in [‘OFF’, ‘SPRAY_FOAM’, ‘SPRAY_BUBBLES’, ‘FOAM_BUBBLES’, ‘SPRAY_FOAM_BUBBLES’], default ‘OFF’

sndparticle_life_max

Highest possible particle lifetime

TYPE:

float in [0, 10000], default 25.0

sndparticle_life_min

Lowest possible particle lifetime

TYPE:

float in [0, 10000], default 10.0

sndparticle_potential_max_energy

Upper clamping threshold that indicates the fluid speed where cells no longer emit more particles (higher value results in generally less particles

TYPE:

float in [0, 1000], default 5.0

sndparticle_potential_max_trappedair

Upper clamping threshold for marking fluid cells where air is trapped (higher value results in less marked cells)

TYPE:

float in [0, 1000], default 20.0

sndparticle_potential_max_wavecrest

Upper clamping threshold for marking fluid cells as wave crests (higher value results in less marked cells)

TYPE:

float in [0, 1000], default 8.0

sndparticle_potential_min_energy

Lower clamping threshold that indicates the fluid speed where cells start to emit particles (lower values result in generally more particles)

TYPE:

float in [0, 1000], default 1.0

sndparticle_potential_min_trappedair

Lower clamping threshold for marking fluid cells where air is trapped (lower value results in more marked cells)

TYPE:

float in [0, 1000], default 5.0

sndparticle_potential_min_wavecrest

Lower clamping threshold for marking fluid cells as wave crests (lower value results in more marked cells)

TYPE:

float in [0, 1000], default 2.0

sndparticle_potential_radius

Radius to compute potential for each cell (higher values are slower but create smoother potential grids)

TYPE:

int in [1, 4], default 2

sndparticle_sampling_trappedair

Maximum number of particles generated per trapped air cell per frame

TYPE:

int in [0, 10000], default 40

sndparticle_sampling_wavecrest

Maximum number of particles generated per wave crest cell per frame

TYPE:

int in [0, 10000], default 200

sndparticle_update_radius

Radius to compute position update for each particle (higher values are slower but particles move less chaotic)

TYPE:

int in [1, 4], default 2

start_point

Start point

TYPE:

`mathutils.Vector` of 3 items in [-inf, inf], default (0.0, 0.0, 0.0), (readonly)

surface_tension

Surface tension of liquid (higher value results in greater hydrophobic behavior)

TYPE:

float in [0, 100], default 0.0

sys_particle_maximum

Maximum number of fluid particles that are allowed in this simulation

TYPE:

int in [0, inf], default 0

temperature_grid

Smoke temperature grid, range 0 to 1 represents 0 to 1000K

TYPE:

float array of 32 items in [-inf, inf], default (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0), (readonly)

time_scale

Adjust simulation speed

TYPE:

float in [0.0001, 10], default 1.0

timesteps_max

Maximum number of simulation steps to perform for one frame

TYPE:

int in [1, 100], default 4

timesteps_min

Minimum number of simulation steps to perform for one frame

TYPE:

int in [1, 100], default 1

use_adaptive_domain

Adapt simulation resolution and size to fluid

TYPE:

boolean, default False

use_adaptive_timesteps

Automatically decide when to perform multiple simulation steps per frame

TYPE:

boolean, default True

use_bubble_particles

Create bubble particle system

TYPE:

boolean, default False

use_collision_border_back

Enable collisions with back domain border

TYPE:

boolean, default False

use_collision_border_bottom

Enable collisions with bottom domain border

TYPE:

boolean, default False

use_collision_border_front

Enable collisions with front domain border

TYPE:

boolean, default False

use_collision_border_left

Enable collisions with left domain border

TYPE:

boolean, default False

use_collision_border_right

Enable collisions with right domain border

TYPE:

boolean, default False

use_collision_border_top

Enable collisions with top domain border

TYPE:

boolean, default False

use_color_ramp

Render a simulation field while mapping its voxels values to the colors of a ramp or using a predefined color code

TYPE:

boolean, default False

use_diffusion

Enable fluid diffusion settings (e.g. viscosity, surface tension)

TYPE:

boolean, default False

use_dissolve_smoke

Let smoke disappear over time

TYPE:

TYPE:
boolean, default False

use_dissolve_smoke_log

Dissolve smoke in a logarithmic fashion. Dissolves quickly at first, but lingers longer.

TYPE:
boolean, default True

use_flip_particles

Create liquid particle system

TYPE:
boolean, default False

use_foam_particles

Create foam particle system

TYPE:
boolean, default False

use_fractions

Fractional obstacles improve and smoothen the fluid-obstacle boundary

TYPE:
boolean, default False

use_guide

Enable fluid guiding

TYPE:
boolean, default False

use_mesh

Enable fluid mesh (using amplification)

TYPE:
boolean, default True

use_noise

Enable fluid noise (using amplification)

TYPE:
boolean, default False

use_slice

Perform a single slice of the domain object

TYPE:
boolean, default False

use_speed_vectors

Caches velocities of mesh vertices. These will be used (automatically) when rendering with motion blur enabled.

TYPE:
boolean, default False

use_spray_particles

Create spray particle system

TYPE:
boolean, default False

use_tracer_particles

Create tracer particle system

TYPE:
boolean, default False

use_viscosity

Simulate fluids with high viscosity using a special solver

TYPE:
boolean, default False

vector_display_type

- `NEEDLE` Needle – Display vectors as needles.
- `STREAMLINE` Streamlines – Display vectors as streamlines.
- `MAC` MAC Grid – Display vector field as MAC grid.

TYPE:
enum in ['NEEDLE', 'STREAMLINE', 'MAC'], default 'NEEDLE'

vector_field

Vector field to be represented by the display vectors

- `FLUID_VELOCITY` Fluid Velocity – Velocity field of the fluid domain.
- `GUIDE_VELOCITY` Guide Velocity – Guide velocity field of the fluid domain.
- `FORCE` Force – Force field of the fluid domain.

TYPE:
enum in ['FLUID_VELOCITY', 'GUIDE_VELOCITY', 'FORCE'], default 'FLUID_VELOCITY'

vector_scale

Multiplier for scaling the vectors

TYPE:
float in [0, 1000], default 1.0

vector_scale_with_magnitude

Scale vectors with their magnitudes

TYPE:
boolean, default False

vector_show_mac_x

Show X-component of MAC Grid

TYPE:
boolean, default True

vector_show_mac_y

Show Y-component of MAC Grid

TYPE:
boolean, default True

vector_show_mac_z

Show Z-component of MAC Grid

TYPE:

boolean, default True

velocity_grid

Smoke velocity grid

TYPE:

float array of 32 items in [-inf, inf], default (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0), (readonly)

velocity_scale

Factor to control the amount of motion blur

TYPE:

float in [0, inf], default 1.0

viscosity_base

Viscosity setting: value that is multiplied by 10 to the power of (exponent*-1)

TYPE:

float in [0, 10], default 1.0

viscosity_exponent

Negative exponent for the viscosity value (to simplify entering small values e.g. 5*10^-6)

TYPE:

int in [0, 10], default 6

viscosity_value

Viscosity of liquid (higher values result in more viscous fluids, a value of 0 will still apply some viscosity)

TYPE:

float in [0, 10], default 0.05

vorticity

Amount of turbulence and rotation in smoke

TYPE:

float in [0, 4], default 0.0

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:

RETURNS:

The class or default when not found.

RETURN TYPE:

type

Inherited Properties

- bpy_struct.id_data

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

References

- FluidModifier.domain_settings

Copyright © Blender Authors
Made with Furo