

[Skip to content](#)

BlendData(bpy_struct)

base class — `bpy_struct`

class `bpy.types.BlendData(bpy_struct)`

Main data structure representing a .blend file and all its data-blocks

actions

Action data-blocks

TYPE:

`BlendDataActions bpy_prop_collection` of `Action`, (readonly)

armatures

Armature data-blocks

TYPE:

`BlendDataArmatures bpy_prop_collection` of `Armature`, (readonly)

brushes

Brush data-blocks

TYPE:

`BlendDataBrushes bpy_prop_collection` of `Brush`, (readonly)

cache_files

Cache Files data-blocks

TYPE:

`BlendDataCacheFiles bpy_prop_collection` of `CacheFile`, (readonly)

cameras

Camera data-blocks

TYPE:

`BlendDataCameras bpy_prop_collection` of `Camera`, (readonly)

collections

Collection data-blocks

TYPE:

`BlendDataCollections bpy_prop_collection` of `Collection`, (readonly)

curves

Curve data-blocks

TYPE:

`BlendDataCurves bpy_prop_collection` of `Curve`, (readonly)

filepath

Path to the .blend file

TYPE:

string, default “”, (readonly, never None)

fonts

Vector font data-blocks

TYPE:

`BlendDataFonts bpy_prop_collection` of `VectorFont` , (readonly)

grease_pencils

Annotation data-blocks (legacy Grease Pencil)

TYPE:

`BlendDataGreasePencils bpy_prop_collection` of `GreasePencil` , (readonly)

grease_pencils_v3

Grease Pencil data-blocks

TYPE:

`BlendDataGreasePencilsV3 bpy_prop_collection` of `GreasePencilv3` , (readonly)

hair_curves

Hair curve data-blocks

TYPE:

`BlendDataHairCurves bpy_prop_collection` of `Curves` , (readonly)

images

Image data-blocks

TYPE:

`BlendDataImages bpy_prop_collection` of `Image` , (readonly)

is_dirty

Have recent edits been saved to disk

TYPE:

boolean, default False, (readonly)

is_saved

Has the current session been saved to disk as a .blend file

TYPE:

boolean, default False, (readonly)

lattices

Lattice data-blocks

TYPE:

`BlendDataLattices bpy_prop_collection` of `Lattice` , (readonly)

libraries

Library data-blocks

TYPE:

`BlendDataLibraries bpy_prop_collection` of `Library` , (readonly)

lightprobes

Light Probe data-blocks

TYPE:

`BlendDataProbes bpy_prop_collection` of `LightProbe` , (readonly)

lights

Light data-blocks

TYPE:

`BlendDataLights bpy_prop_collection of Light`, (readonly)

linestyles

Line Style data-blocks

TYPE:

`BlendDataLineStyles bpy_prop_collection of FreestyleLineStyle`, (readonly)

masks

Masks data-blocks

TYPE:

`BlendDataMasks bpy_prop_collection of Mask`, (readonly)

materials

Material data-blocks

TYPE:

`BlendDataMaterials bpy_prop_collection of Material`, (readonly)

meshes

Mesh data-blocks

TYPE:

`BlendDataMeshes bpy_prop_collection of Mesh`, (readonly)

metaballs

Metaball data-blocks

TYPE:

`BlendDataMetaBalls bpy_prop_collection of MetaBall`, (readonly)

movieclips

Movie Clip data-blocks

TYPE:

`BlendDataMovieClips bpy_prop_collection of MovieClip`, (readonly)

node_groups

Node group data-blocks

TYPE:

`BlendDataNodeTrees bpy_prop_collection of NodeTree`, (readonly)

objects

Object data-blocks

TYPE:

`BlendDataObjects bpy_prop_collection of Object`, (readonly)

paint_curves

Paint Curves data-blocks

TYPE:

FILE:

`BlendDataPaintCurves bpy_prop_collection` of `PaintCurve` , (readonly)

palettes

Palette data-blocks

TYPE:

`BlendDataPalettes bpy_prop_collection` of `Palette` , (readonly)

particles

Particle data-blocks

TYPE:

`BlendDataParticles bpy_prop_collection` of `ParticleSettings` , (readonly)

pointclouds

Point cloud data-blocks

TYPE:

`BlendDataPointClouds bpy_prop_collection` of `PointCloud` , (readonly)

scenes

Scene data-blocks

TYPE:

`BlendDataScenes bpy_prop_collection` of `Scene` , (readonly)

screens

Screen data-blocks

TYPE:

`BlendDataScreens bpy_prop_collection` of `Screen` , (readonly)

shape_keys

Shape Key data-blocks

TYPE:

`bpy_prop_collection` of `Key` , (readonly)

sounds

Sound data-blocks

TYPE:

`BlendDataSounds bpy_prop_collection` of `Sound` , (readonly)

speakers

Speaker data-blocks

TYPE:

`BlendDataSpeakers bpy_prop_collection` of `Speaker` , (readonly)

texts

Text data-blocks

TYPE:

`BlendDataTexts bpy_prop_collection` of `Text` , (readonly)

textures

Texture data-blocks

Texture data-blocks

TYPE:

`BlendDataTextures bpy_prop_collection` of `Texture`, (readonly)

use_autopack

Automatically pack all external data into .blend file

TYPE:

boolean, default False

version

File format version the .blend file was saved with

TYPE:

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

volumes

Volume data-blocks

TYPE:

`BlendDataVolumes bpy_prop_collection` of `Volume`, (readonly)

window_managers

Window manager data-blocks

TYPE:

`BlendDataWindowManagers bpy_prop_collection` of `WindowManager`, (readonly)

workspaces

Workspace data-blocks

TYPE:

`BlendDataWorkSpaces bpy_prop_collection` of `WorkSpace`, (readonly)

worlds

World data-blocks

TYPE:

`BlendDataWorlds bpy_prop_collection` of `World`, (readonly)

batch_remove(ids)

Remove (delete) several IDs at once.

Note that this function is quicker than individual calls to `remove()` (from `bpy.types.BlendData` ID collections), but less safe/versatile (it can break Blender, e.g. by removing all scenes...).

PARAMETERS:

ids (Sequence[`bpy.types.ID`]) – Sequence of IDs (types can be mixed).

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

file_path_map(subset=None, key_types=None, include_libraries=False)

Returns a mapping of all ID data-blocks in current `bpy.data` to a set of all file paths used by them.

For list of valid set members for `key_types`, see: `bpy.types.KeyingSetPath.id_type`.

PARAMETERS:

- **subset** (*sequence*) – When given, only these data-blocks and their used file paths will be included as keys/values in the map.
- **key_types** (*set of strings*) – When given, filter the keys mapped by ID types. Ignored if `subset` is also given.
- **include_libraries** (*bool*) – Include library file paths of linked data. False by default.

RETURNS:

dictionary of `bpy.types.ID` instances, with sets of file path strings as their values.

RETURN TYPE:

dict

orphans_purge()

Remove (delete) all IDs with no user.

PARAMETERS:

- **do_local_ids** (*bool, optional*) – Include unused local IDs in the deletion, defaults to True
- **do_linked_ids** (*bool, optional*) – Include unused linked IDs in the deletion, defaults to True
- **do_recursive** (*bool, optional*) – Recursively check for unused IDs, ensuring no orphaned one remain after a single run of that function, defaults to False

RETURNS:

The number of deleted IDs.

temp_data(filepath=None)

A context manager that temporarily creates blender file data.

PARAMETERS:

filepath (*str | bytes | None*) – The file path for the newly temporary data. When None, the path of the currently open file is used.

RETURNS:

Blend file data which is freed once the context exists.

RETURN TYPE:

`bpy.types.BlendData`

user_map(subset, key_types, value_types)

Returns a mapping of all ID data-blocks in current `bpy.data` to a set of all data-blocks using them.

For list of valid set members for `key_types` & `value_types`, see: `bpy.types.KeyingSetPath.id_type`.

PARAMETERS:

- **subset** (*Sequence[`bpy.types.ID`]*) – When passed, only these data-blocks and their users will be included as keys/values in the map.
- **key_types** (*set[str]*) – Filter the keys mapped by ID types.

- **value_types** (*set[str]*) – Filter the values in the set by ID types.

RETURNS:

dictionary that maps data-blocks ID's to their users.

RETURN TYPE:

dict[bpy.types.ID, set[bpy.types.ID]]

Inherited Properties

- `bpy_struct.id_data`

Inherited Functions

- | | |
|---|--|
| • <code>bpy_struct.as_pointer</code> | • <code>bpy_struct.items</code> |
| • <code>bpy_struct.driver_add</code> | • <code>bpy_struct.keyframe_delete</code> |
| • <code>bpy_struct.driver_remove</code> | • <code>bpy_struct.keyframe_insert</code> |
| • <code>bpy_struct.get</code> | • <code>bpy_struct.keys</code> |
| • <code>bpy_struct.id_properties_clear</code> | • <code>bpy_struct.path_from_id</code> |
| • <code>bpy_struct.id_properties_ensure</code> | • <code>bpy_struct.path_resolve</code> |
| • <code>bpy_struct.id_properties_ui</code> | • <code>bpy_struct.pop</code> |
| • <code>bpy_struct.is_property_hidden</code> | • <code>bpy_struct.property_overridable_library_set</code> |
| • <code>bpy_struct.is_property_overridable_library</code> | • <code>bpy_struct.property_unset</code> |
| • <code>bpy_struct.is_property_readonly</code> | • <code>bpy_struct.type_recast</code> |
| • <code>bpy_struct.is_property_set</code> | • <code>bpy_struct.values</code> |

References

- `Context.blend_data` • `RenderEngine.update`