3D Digital Backbone Driver for Blender

USER MANUAL

TABLE OF CONTENTS

This documentation explains how to install and work with the **3D Digital Backbone Driver add-on for Blender** by LUMISCAPHE, enabling seamless integration with the 3D Digital Backbone. It applies to version 1.0.0.

Version: 20251029.1057

Headquarters:

ZA du Courneau II
8, impasse de la rouilleuse
33610 Canéjan
France

Documentation lead: P. Peyrevidal

INTRODUCTION

3D Digital Backbone Driver for Blender

This documentation explains how to install and work with the **3D Digital Backbone Driver add-on for Blender** by LUMISCAPHE, enabling seamless integration with the 3D Digital Backbone. It applies to version 1.0.0.

The **3D Digital Backbone Driver for Blender** is an extension/add-on that connects it to the 3D Digital Backbone.

> **NOTE**
> This software and its resources are available at https://patchwork3d.com/3d-digital-backbone-tools.

The 3D Digital Backbone

The 3D Digital Backbone connects your software and 3D data via a customized infrastructure to orchestrate your production processes according to your business needs, without disrupting your digital environment.

The connection to the 3D Digital Backbone gives some advantages:

- **Historization of the data**
  Each save is recorded in the 3D Digital Backbone. It is possible to enable/disable any actions saved in the database.
- **Collaboration**
  Synchronize data between multiple instances of Blender, and collaborate on the same model.
- **Interop**
  Convert Blender data to other formats using 3D Digital Backbone converters.

> **NOTE**
> To learn more about 3D Digital Backbone: https://lumiscaphe.com/en/digital-backbone-2/

License

The add-on source code is LGPL licensed by LUMISCAPHE (lumiscaphe.com).

The add-on embeds a DS Viper Python runtime wheel component licensed under an Open License from DIGITAL SUBSTRATE (digitalsubstrate.io).

By using the add-on, you agree to comply with these licenses, the texts of which are available in the add-on folder once it has been installed.

INSTALLATION

This part explains the system requirements and provides instructions to install and activate the add-on in Blender.

Requirements

Operating system:

- Windows 11 x86
- macOS 15 Apple Silicon
- Linux x86 with glibc 2.28 and greater library: Debian 10+, Ubuntu 18.10+, Fedora 29+, CentOS/RHEL 8+

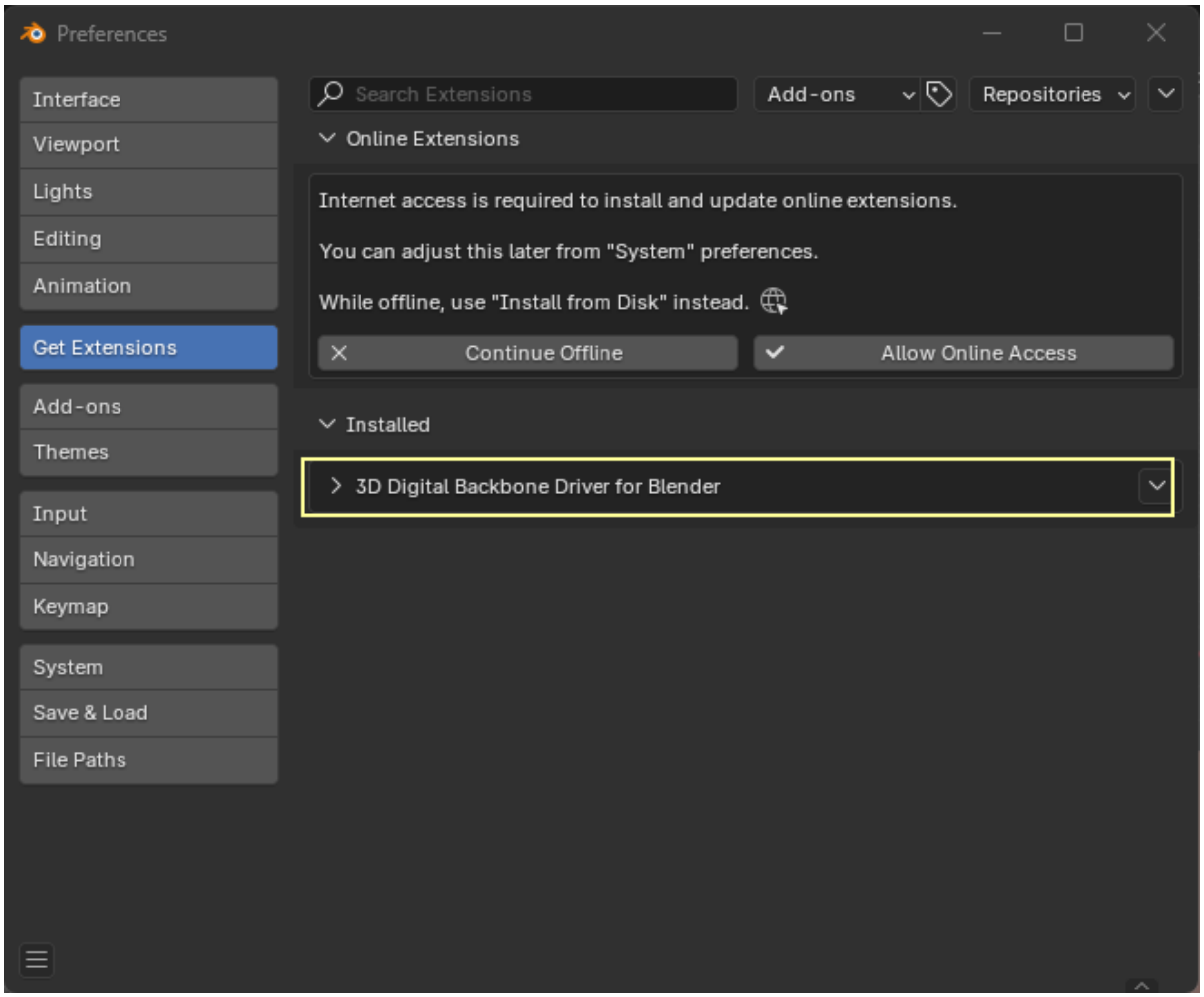
Blender:

Blender 4.5 is recommended, but 4.3 and 4.4 are also supported.

Installation guide

The **3D Digital Backbone Driver** add-on is provided in a .zip file and can be installed from disk by following these instructions: https://docs.blender.org/manual/en/4.3/editors/preferences/extensions.html.

Once installed, the add-on should appear in the list of installed extensions and be activated as an add-on in the menu ***Edit*** > ***Preferences*** > ***Add-ons*** tab.



CONCEPTS

This part defines the key terms and concepts used in the documentation and details the types of data supported by the add-on.

Vocabulary

In this documentation, the following concepts will be used several times:

- **.blend** file / **Blender data** file
  The native file format used by Blender to save/load data.
- **3D Digital Backbone**, simplified as **DBB**
  Database used to store the Blender database, which corresponds to the content of a .blend file. A local database file is used in this documentation. The file extension can be anything; in this documentation, .dbb is used. A DBB can contain multiple Blender databases, meaning that in Blender, when loading content from a DBB you have to first select the filepath and then select the name of the Blender database to load, it is a two-step process.
- **Database name**
  A name associated with the Blender database saved in the DBB, as multiple Blender databases can be saved in one DBB, the name acts as the identifier for the Blender database. This name is chosen by the user.

Supported data

The following properties in each data block are saved:

- **Scene**: name | collection children | objects children
- **Collections**: name | colors | hide_status | collection children | objects children
- **Objects**: name | data binding | transform | parent | hide_status
- **Meshes**: name | mesh essential data (vertices/edges/loops/polygons)
- **Materials**: only nodal materials and the nodes Output and BSDFPrinciple
- **Light**: name and for Point Light, the power property
- **Images**: all properties

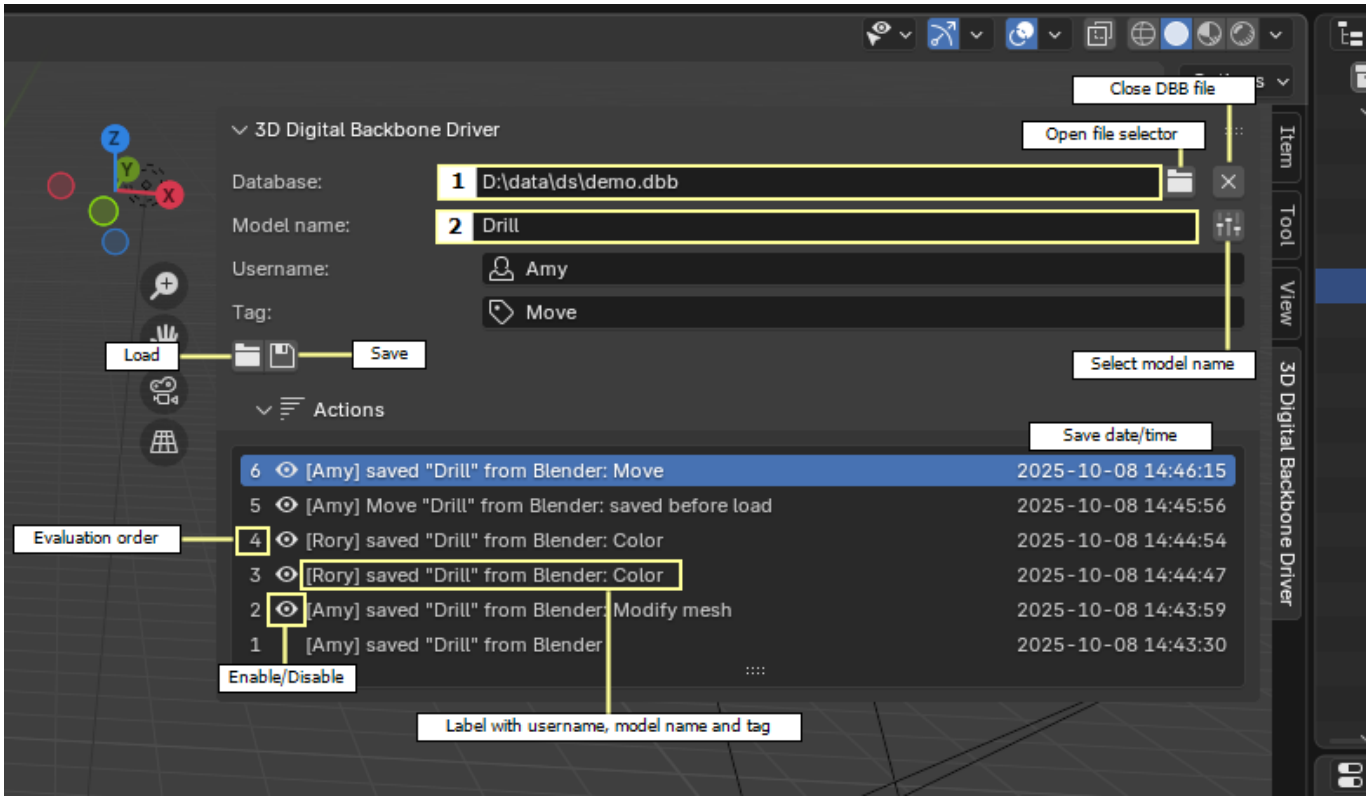
NOTE
For cameras, no specific properties are saved. Only its properties as objects are saved.

ADD-ON USAGE

This part describes how to use the add-on within Blender, including the user interface, version management, and the main operations for saving, loading, and synchronizing data.

Using the User Interface

This section introduces the add-on's interface in Blender and explains how to navigate its main features, such as selecting data, customizing version labels, and managing version history.



How to select the Blender data in the DBB?

1. Click on the file selector in the first row to choose the 3D Digital Backbone database file to load: select an existing file in which the content will be saved. If the file does not exist, it will be created at the first save operation.
2. Choose a Blender database name:
   - If it is the first save, write a name in the Name field.
   - If it is a model already saved in the DBB, click on to display the names of the models stored in the DBB.

Customizing the version label

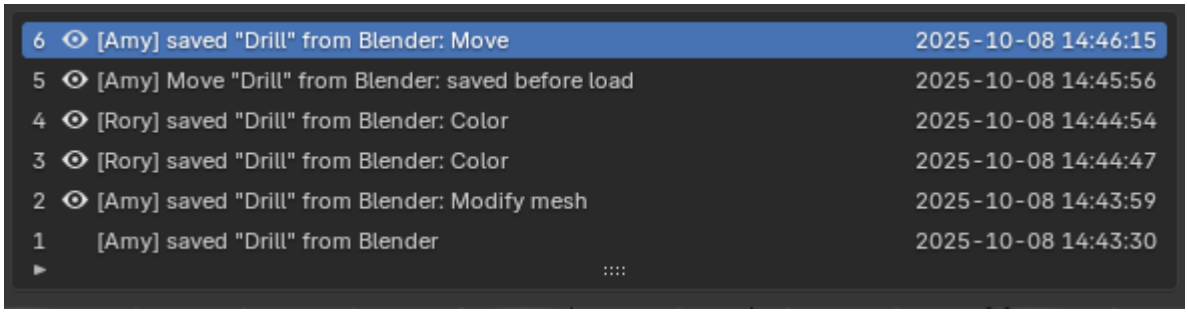
Each time the save button is pressed, the current state of the data in Blender will be saved in the DBB along with a label composed of the username and tags. In the interface, by default, the username is set to the username of the computer. If the username field is left empty, in the version label, it will be mentioned "***Anonymous user***".

The optional tag can be helpful to identify the version.

List of actions

This list allows the user to understand, for a loaded state, the order in which actions were evaluated to compute the current state. By toggling the eye icon the user can enable or disable a state, and the result will be automatically updated in the 3D Digital Backbone and in the opened Blender session.

For example, the list of actions shown below.



In order of evaluation, the actions evaluated are:

- User [Amy] saved the data "Drill" from Blender.
- User [Amy] saved Blender's modified "Drill" data with the tag [Modify mesh].
- User [Rory] saved Blender "Drill" data with the tag [Color].
- User [Rory] saved Blender "Drill" data with the tag [Color].
- User [Amy] tried to load, so her work was saved first.
- User [Amy] saved Blender's modified "Drill" data with the tag [Move].

Working with the DBB

This section provides step-by-step procedures for initializing, saving, loading, synchronizing, and closing Blender data in the 3D Digital Backbone.

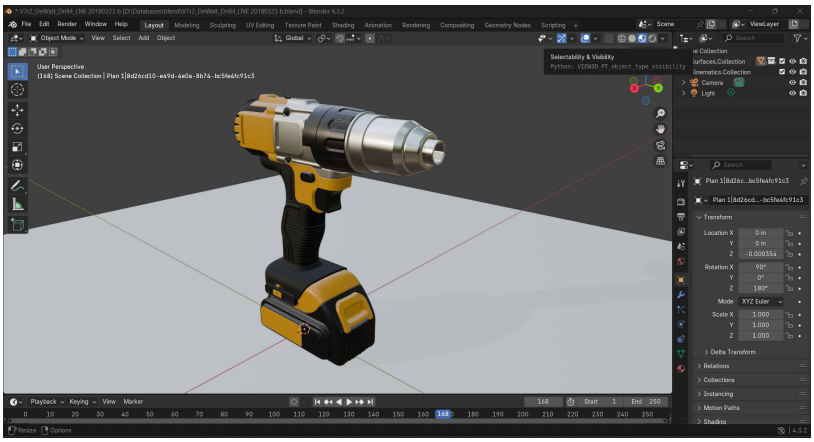
For databases containing a lot of meshes, the save and initial load operations can take some time.

How to initialize the Blender data representation in DBB from a .blend file?

Context: The data is not in the DBB, and you want to save it.

1. Load a .blend file
2. Select a DBB database path and model name [8].
3. [Optional] Set a user name and tag keyword.
4. Click on the ***Save*** button.

.blend →

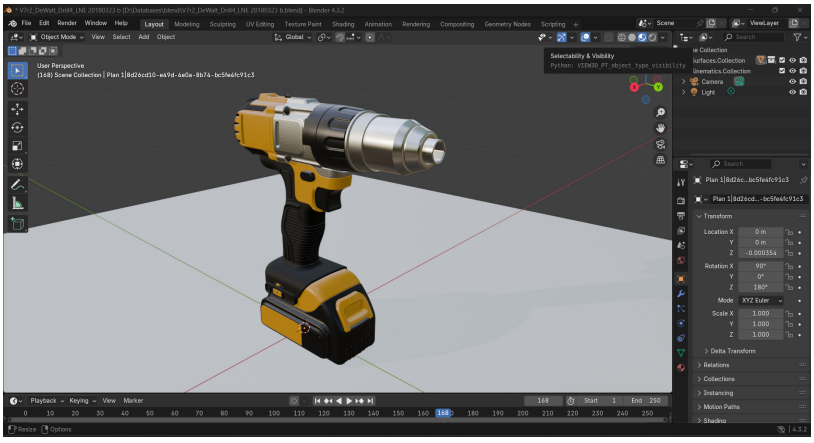


→ DBB

How to save the new modifications to the DBB?

Context: The data is already saved to the DBB, and you want to save new modifications.

1. Select a DBB database path and model name [8].
2. [Optional] Set a user name and tag keyword.
3. Click on the ***Save*** button.



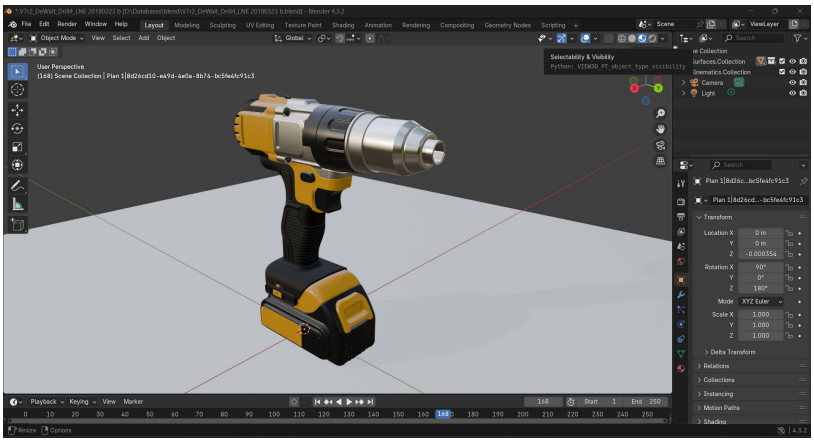
→ DBB

How to load the Blender database from the DBB?

> **WARNING**
> Loading a DBB will clear all data from the current Blender session. It is safer to start from an empty Blender session.

Context: The data is already saved to the DBB, and you want to load it into your current Blender session.

1. Open Blender without loading a specific .blend file.
2. Select a DBB database path and model name [8].
3. Click on the ***Save*** button.
4. Your Blender is ready to be used!



← DBB

How to synchronize the local Blender data with the Blender data in the DBB?

Prerequisites: Blender database is loaded from the DBB [11].

Context: The data is already saved to the DBB, you want to save all your modifications and merge them with all the recent modifications done by other users.

Choosing to click on the ***Synchronize*** button means that your work will be saved, then all the modifications on the database will be merged, and the result of it will be displayed in your current Blender session.

How to close a DBB file?

Prerequisites: Blender database is loaded from the DBB [11].

Context: This functionality may be required if you want to delete the DBB file, but the database is open in your current Blender session: if you click ***Close***, you can close the database without closing Blender.

Click on the ***Close*** button. The DBB file has to be opened previously; otherwise, the button is disabled.