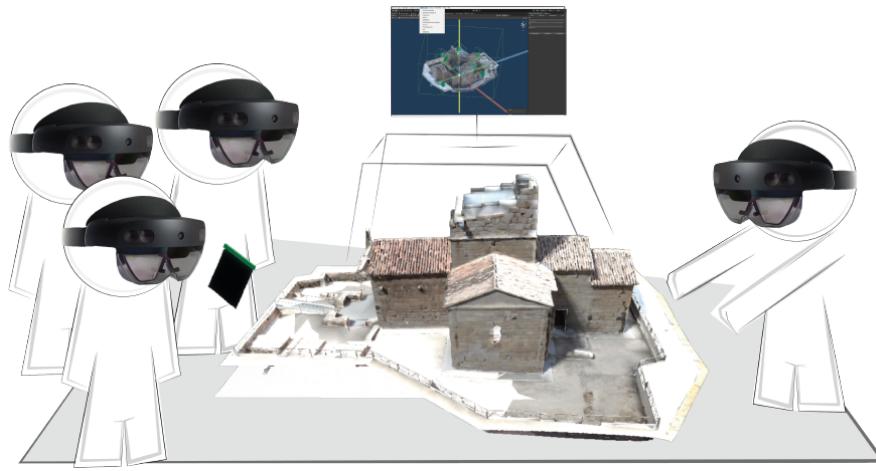


# Learn Spatial!

# Introducing the MARBLE App



## Introducing MARBLE

MARBLE is an interactive multi-user presentation platform for displaying architecture on the HoloLens2. With MARBLE one can create content packages (seminars) and display them on multiple HL2 devices. MARBLE displays the 3D content for everybody at the same position with the same orientation to enable a shared learning experience. Additional MARBLE is packed with "tools" to interact with the presented 3D model, like pointers or clipping planes.

## MARBLE Background

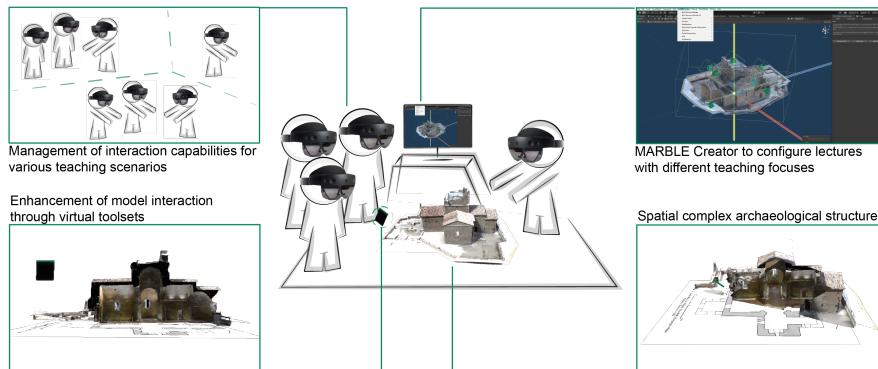
MARBLE is a cooperative project between the Albert-Ludwigs-University Freiburg and Furtwangen University. It explores the possibility of enriching archaeological higher education using mixed reality technology. It is based on the idea of offering students the opportunity to perceive and examine excavations and artefacts in three dimensions without having to travel to excavations. Instead of having to continue working with two-dimensional sources such as photographs or videos, MARBLE offers a shared three-dimensional interactive experience for a whole group of students.



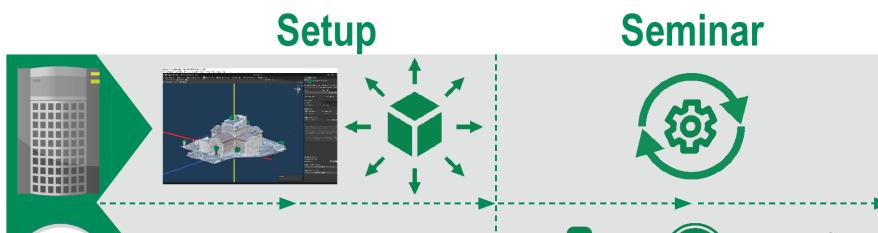
# Learn »A Mixed Reality Approach to Enhance Archaeological Higher Education«!

## A Mixed Reality Approach to Enhance Archaeological Higher Education

The concept behind MARBLE is to offer a Mixed Reality platform for non-tech-savvy teaching stuff and students. In general, the MARBLE-App consists of two parts. One part is the MARBLE Creator, which is implemented as a Unity Editor integration for teachers and serves as a tool for preparing MARBLE sessions. The other part is the MARBLE Player, the presentation tool that runs on the Microsoft's HoloLens2 (HL2) and displays the main model prepared with the Creator. Like the Creator, the Player is also developed with Unity. The overview image shows the main concept and features of MARBLE.



As it is known that AR can create cognitive overload on students, which can negatively affect educational effectiveness [1], the development of the MARBLE-App is subject to the paradigms of user-friendliness and the reduction of setup effort to a minimum for a non-tech-savvy target group. The students should be able to dive straight into the course without having to laboriously set up the Player each time. The following picture illustrates the base interaction between the Creator and the Player. The Creator on the top row is for preparing and setting up the following seminar, in which the Creator synchronizes all clients. The Player first requests the current seminar, locates its spatial position, and then displays the seminar's model.



# Learn »Install & Setup«!

## Install & Setup

MARBLE and the current version are experimental research versions. There is no guarantee that they will properly work and may cause a lot of troubleshooting to get it to work. MARBLE depends on a lot of settings, plugins and third party libraries. The following guide will lead you through the most important installation steps. The core of the MARBLE app is the current package »marble\_mordor\_1.0.0« which you have to [request from the author](#). If you have access to the unity package, follow this guide (chronologically) to run MARBLE on your machine.

## Visual Studio

To properly deploy new app versions, you need to follow the instructions from: [Microsoft Learn - Mixed Reality](#).

## Unity

### Version

MARBLE runs on 2020.3.13f1. Make sure to also install the UWP module.

### Packages

The latest version of MARBLE may not require all the following packages. To enable better troubleshooting, they are listed below:

- Editor Coroutines 1.0.0
- OpenXR Plugin 1.3.1
- TextMeshPro 3.0.6
- Windows XR Plugin 4.5.0
- XR Plugin Management 4.2.1

### Mixed Reality Toolkit

MARBLE was developed with MRTK 2.7.3. Make sure to have the following packages:

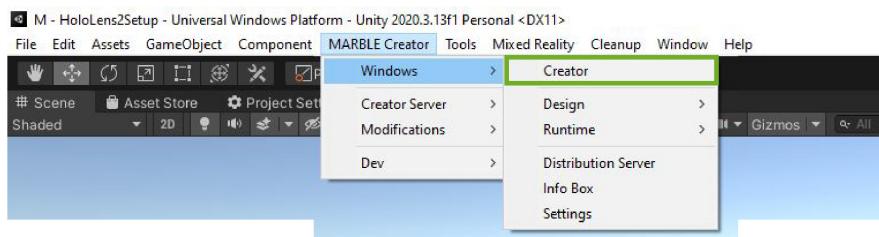
# Learn »Getting Started«!

## Getting Started

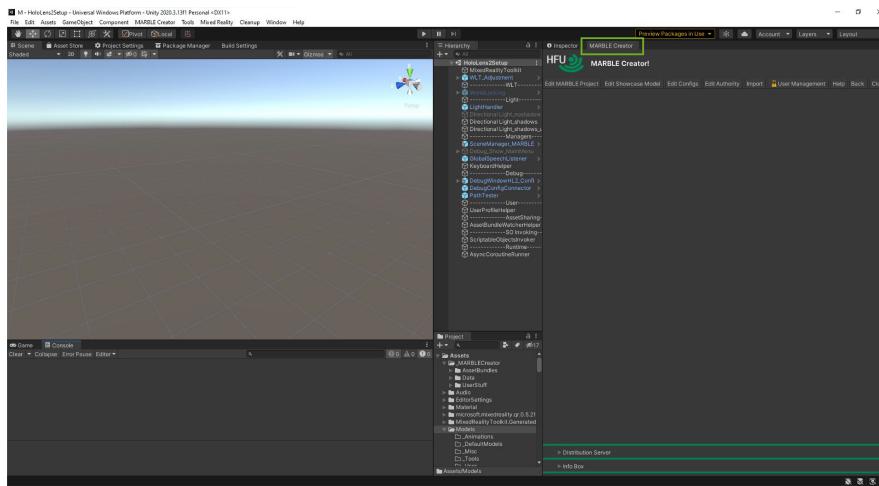
First of all make sure that you have installed MARBLE as stated in [Install & Setup](#).

## Open MARBLE Creator

If the Creator panel does not pop up automatically, call it from the top menu. The figure illustrates that you can find the Creator panel under "MARBLE Creator > Windows > Creator":



As the Creator panel is a simple unity panel, you can dock it as any other panel. See the next figure:



## Create A Project

MARBLE is based on MARBLE Projects. Those are data structures which contain and reference all important data for your session. In order to use MARBLE synchronized on the HoloLens2 devices, you need at least one MARBLE Project which you distribute as package (see [Concept & Architecture - AssetBundles And Updates](#)). To create a MARBLE Project, enter a unique name and click

# Learn »Concept & Architecture«!

## Concept & Architecture

In order to use the Creator as it is intended, it is useful to understand the basics about the concept and architecture of the software.

## Configurations & Asset Updates

### Layers of Configurations

The control of the functions and the way they are accessed vary from function to function. Basically, there are two different levels of settings. The first and lowest is the "**Config**" level. The options to be set via the "Config" files determine which features can be controlled and used in the session at all. This also includes, for example, which features are displayed in the user's hand menu. Features that are not enabled via the "Configs" cannot be controlled by the software at runtime. The second level is the **authority level**, which should not be confused with the network authority of synchronized objects. The authority level determines for each profile which of the features may be used by the user of the profile. It is essential that the features are also authorized by the "Config" level. If the "Config" level prohibits a feature, it will not be made available regardless of the authority level. This means that every available feature in a session must be activated in the "Configs" in order to be used. If a profile is to use a feature that is activated in the "Configs", its authorization level must be adjusted.

### Asset Bundles and Updates

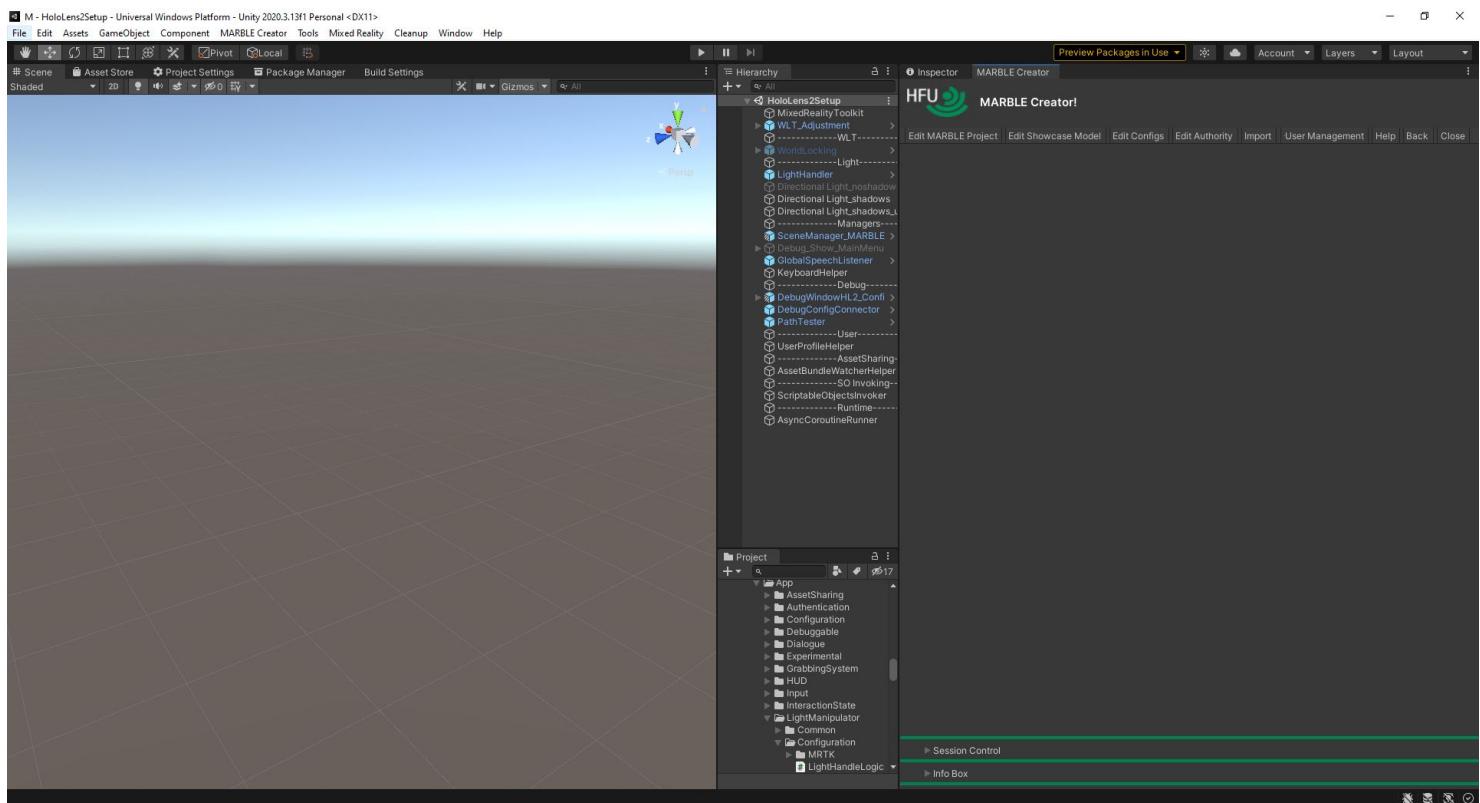
In MARBLE, sessions are represented as "asset bundles". These bundles contain all the necessary information and models for a session. They also contain configurations and authorization levels for each individual client. Each session can therefore have its own configurations and authorization levels depending on its area of use. By default, each client requests the current asset bundle from the server and updates all information (e.g. configurations, permission levels, settings, models, etc.) based on these bundles. As a result, each session on each client has the same settings and data. However, it is possible to bypass this behaviour and inject clients with individual configurations and settings. To do this, it is necessary to understand MARBLE's asset update process. When the MARBLE Player is started, it first loads the configuration files onto the respective systems. This is important to avoid unwanted session settings from the asset bundles. After loading these configurations, the update process checks which data should be requested. The "requestAnchor", "requestConfig" and "requestAssets" configs of the "NetworkConfig" or "netconfig.txt" file determine which contents are to be requested (currently only the asset bundles can be requested). After the data has been requested and hopefully received, the update process checks which of this data should actually be used in the current session. Important for this process are the configs "updateIp", "updateConfigs" and "updateAssets".

### Irreversible Bypassing

# Learn »Unity and Creator Overview«!

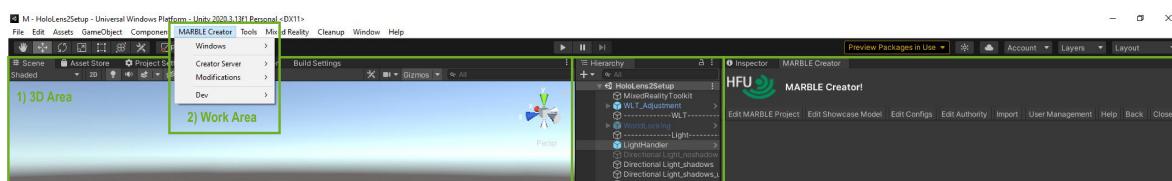
## Unity and Creator Overview

The following image shows Unity and the Creator Plugin.



## Unity Editor Areas

1. 3D Area --> Shows the current scene. Will display the model when opening "Edit Showcase Model".
2. Creator Menu Item --> Menu navigation to open Creator and its features.
3. Scene Graph --> All elements in the scene. Or model components when "Edit Showcase Model" is open.
4. Project Graph --> Folders of the Unity Project.
5. Creator --> The actual MARBLE Creator plugin window.



# Learn »Import Models«!

## Import Models

The Import panel provides an easy way to import supported models into the Creator and thus into Unity.

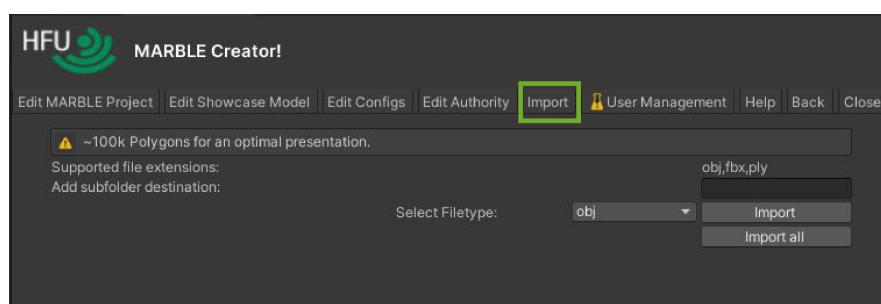
### Navigate to import

Navigate to the "Import" menu item in the Creator. A window opens where you can import models to use with MARBLE (see below). You will see the following labels and options:

### Overview

LABEL	CONTENT
<b>polygon count warning</b>	For the best and most stable results, only use models with around 100k polygons
<b>supported file extensions</b>	At the moment, you can only import .obj models with the importer. All other unity supported files have to be imported manually.
<b>add subfolder destination</b>	Use this to create a folder in which all the models will be imported to. The models will still be independently imported in their own subfolder.
<b>Import</b>	Will import the selected .obj either to the default import path or the default import path + subfolder.
<b>Import all</b>	Will get all .obj files of the selected directory and import them either to the default import path or the default import path + subfolder.

Be aware that this can take some time. After successful import, the models can be integrated into the seminars.



# Learn »MARBLE Projects«!

## MARBLE Projects

The heart of a MARBLE session (or seminar/ lecture) are the MARBLE projects. In a MARBLE project, all necessary information is gathered and can be exported as `_AssetBundle_`.

## MARBLE Project Content

A MARBLE Project consists of the following components:

COMPONENT	DESCRIPTION
<b>Author Name</b>	Name of the author.
<b>Project Name</b>	Name of the project. <span style="color: orange;">⚠</span> Be aware that changing this name also changes the project's file name.
<b>Selected Models</b>	All shown models for the project/ session are managed in a database. Add a specific model database to the project to be able to show its models during a session.
<b>Selected Tools</b>	All usable tools in the project/ session are managed in a database. Add a specific tool database to the project to be able to use its tools during a session.
<b>Authority Settings</b>	Add a database for all the authority settings which are used to determine what the user is allowed to do.
<b>Config Files</b>	Configuration files can be used to configure different areas of the application depending on the use case of the session.

- Main Config - General settings, e.g. whether to skip the main menu.
- Network Config - Network-related settings like ip-adress of the server or asset update parameters.
- Menu Config - Which features should be shown in the main and hand menus.
- Feature Config - Which features should be enabled during the session.
- Pointer Config - How pointers are presented during the session, e.g. whether they are enabled from the beginning of the session for everyone.

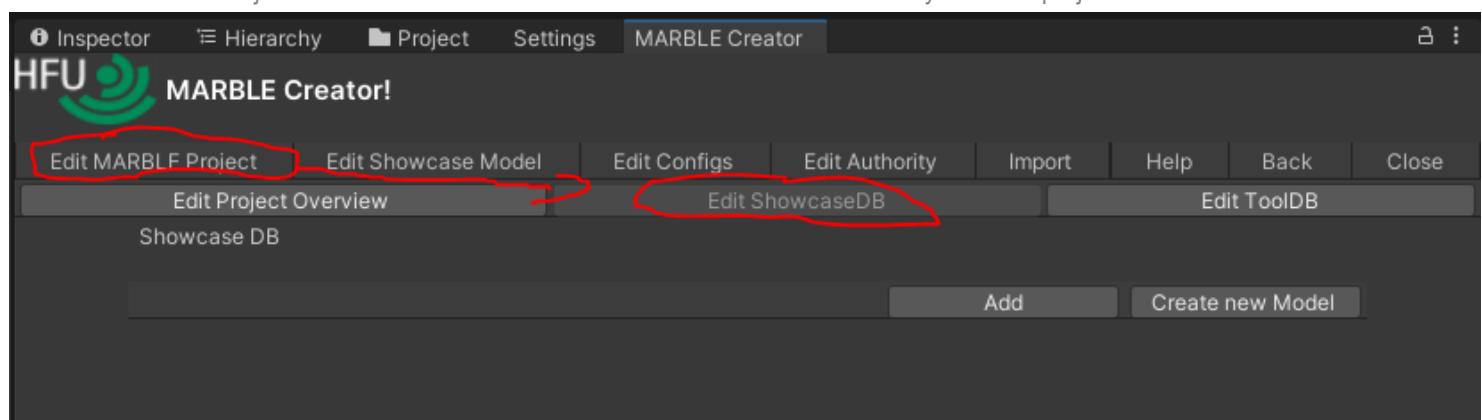
# Learn »Showcase Databases«!

## Showcase Databases

A showcase database is a collection of models which can be used in a session/ seminar and are packed into a package when exporting a project. Basically you can add as many models as you want, but be aware that this can cause long export times and also lead to memory exceptions.

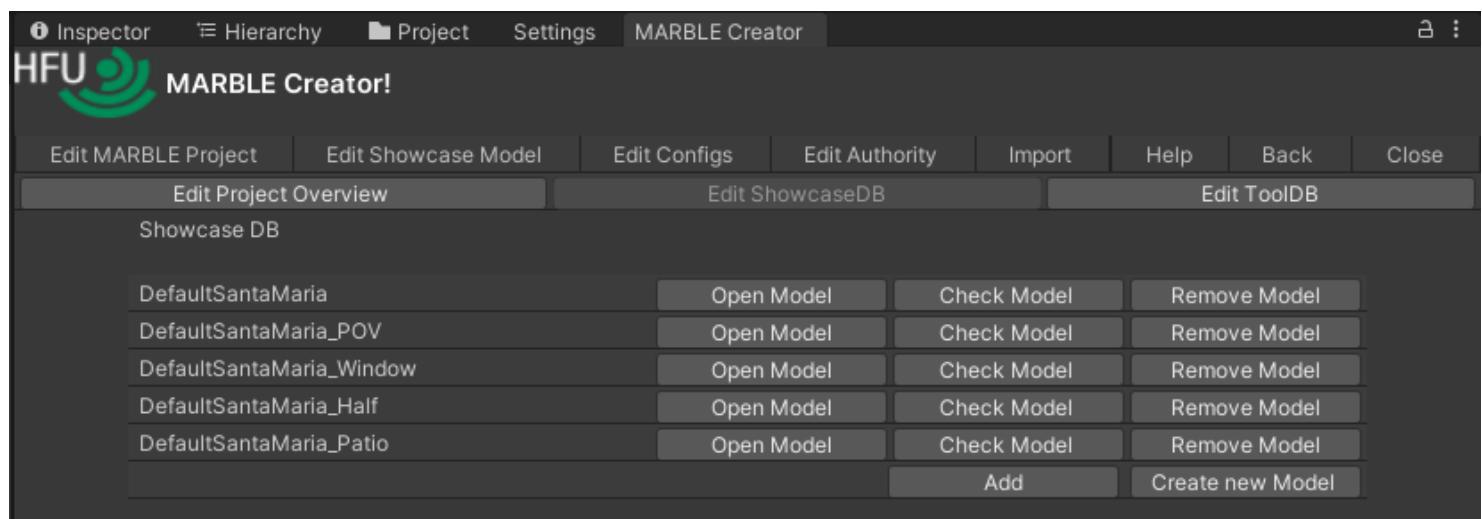
## Navigation

Via "Edit MARBLE Project" > "Edit ShowcaseDB" the model database of the currently selected project can be edited.



## Add & Remove Models

By clicking on "Add" models can be added to the database. When added also all "submodels" will be added automatically if they aren't already added.



The screenshot shows the "Edit ShowcaseDB" screen. At the top is a navigation bar with tabs: Inspector, Hierarchy, Project, Settings, and MARBLE Creator. Below the navigation bar is a toolbar with buttons: Edit MARBLE Project, Edit Showcase Model, Edit Configs, Edit Authority, Import, Help, Back, and Close. Underneath the toolbar is a secondary row of buttons: Edit Project Overview, Edit ShowcaseDB (highlighted with a red box), and Edit ToolDB. The main workspace displays a table of models in the "Showcase DB". The table has four columns: Model Name, Open Model, Check Model, and Remove Model. The models listed are: DefaultSantaMaria, DefaultSantaMaria\_POV, DefaultSantaMaria\_Window, DefaultSantaMaria\_Half, and DefaultSantaMaria\_Patio. At the bottom of the workspace are two buttons: Add and Create new Model.

Model Name	Open Model	Check Model	Remove Model
DefaultSantaMaria	Open Model	Check Model	Remove Model
DefaultSantaMaria_POV	Open Model	Check Model	Remove Model
DefaultSantaMaria_Window	Open Model	Check Model	Remove Model
DefaultSantaMaria_Half	Open Model	Check Model	Remove Model
DefaultSantaMaria_Patio	Open Model	Check Model	Remove Model

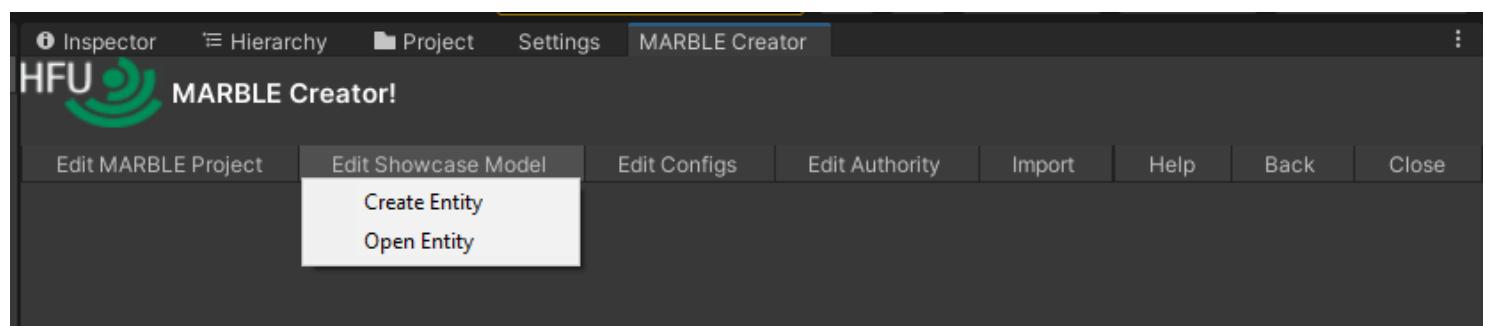
# Learn »Showcase Models«!

## Showcase Models

A showcase model is the data structure wrapped around a model file. The showcase model data structure holds all the necessary data for a displaying and connecting a model during runtime.

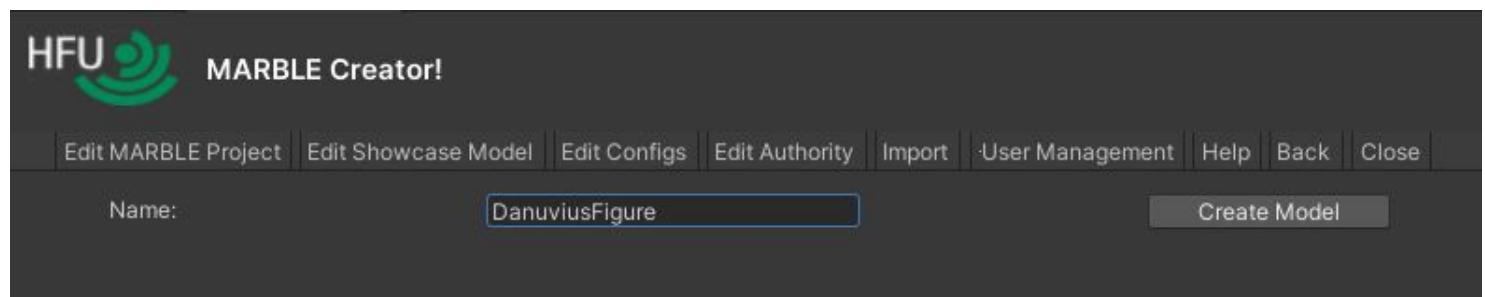
## Navigation

The regular way to edit a model is via the menu bar. Go to "Edit Showcase Model" and either select "Create Entity" or "Open Entity" to either create new showcase model or open an already existing one.



## Create Model

To create a model just select "Edit Showcase Model" > "Create Entity". Enter a name and click "Create Model". MARBLE will open an empty model structure for you to configure.



## Open Model

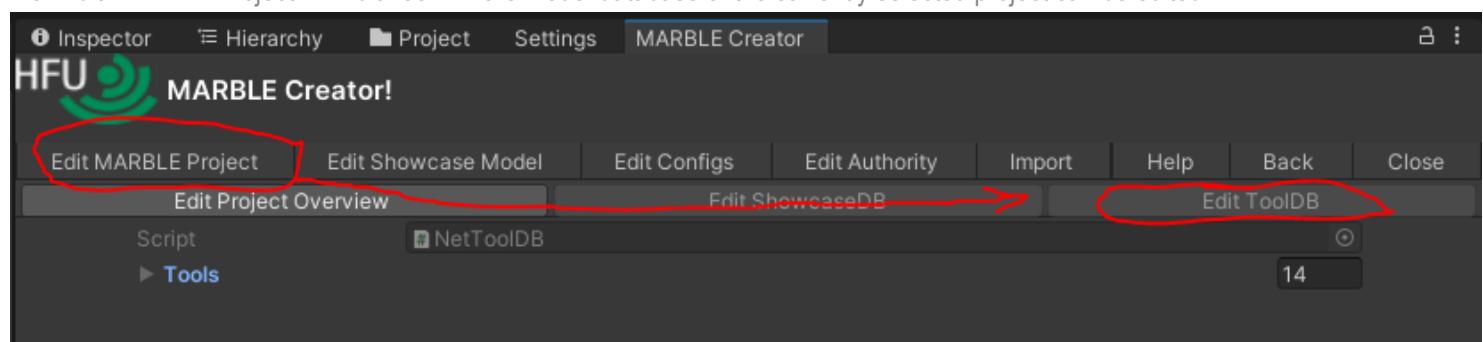
# Learn »Tool Databases«!

## Tool Databases

A tool database is a list holding all tools which should be available during a session. Basically you can add as many tools as you want, but be aware that this can cause long export times and also lead to memory exceptions.

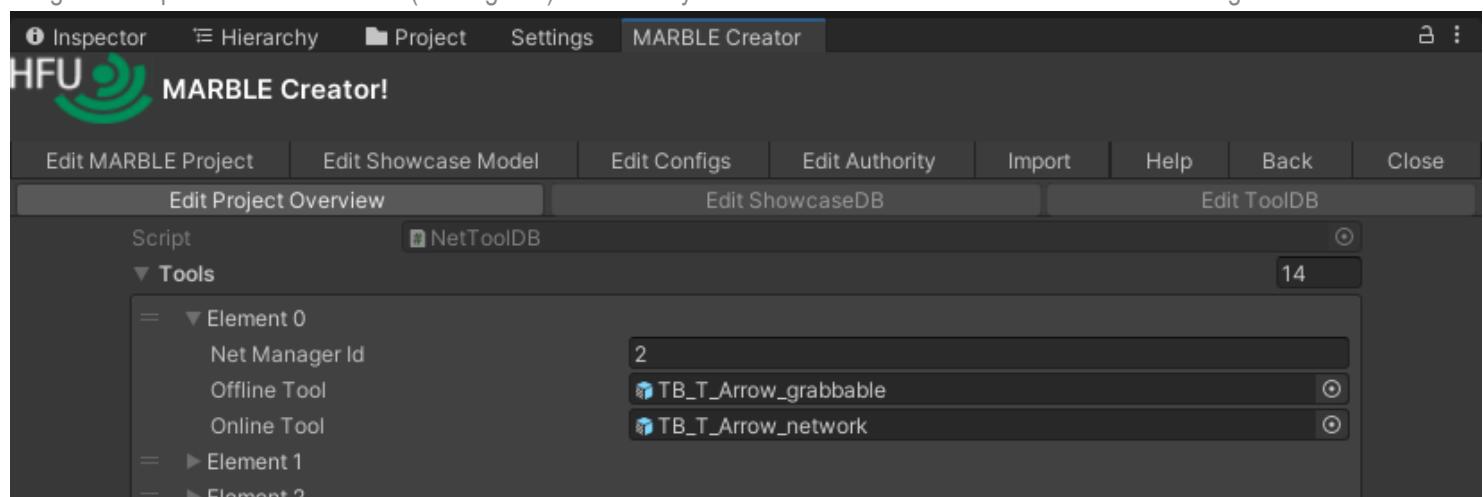
## Navigation

Via "Edit MARBLE Project" > "Edit ToolDB" the model database of the currently selected project can be edited.



## Add & Remove Tools

Currently, adding tools is implemented, but not supported in a user-friendly manner. In order to add a new tool, you have to create a new tool by deriving a new prefab from the "TB\_Tool\_grabbable\_network" and one from "TB\_Tool\_grabbable". In prefab mode, you have to add the desired model and functionality to both prefabs! After this process, you can add a new tool to the tool database and reference the derived "TB\_Tool\_grabbable" in the "Offline Tool" field and the derived "TB\_Tool\_grabbable\_network" in the "Online Tool" field. Further, you have to find the "MARBLE\_NetworkManager" in the project panel, which controls the network session at runtime. Add the online representative of the new tool to the "Registered Spawnable Prefabs" list. Finally, add the position of the new tool in the "Registered Spawnable Prefabs" list (starting at 0) to the newly added tool in the tool database under "Net Manager Id".



# Learn »Configs & Effects«!

## Configs & Effects

Italic configs are currently not implemented. Monospaced should not be changed.

### Main Config

IDENTIFIER	EXPECTED BEHAVIOUR
autospawn	Leave <code>false</code> - Don't change
keepQRtrack	Leave <code>false</code> - Don't change
useLibrary	Leave <code>true</code> - Don't change
rotateMX	Determines if the model can be rotated around X-axis.
rotateMY	Determines if the model can be rotated around Y-axis.
rotateMZ	Determines if the model can be rotated around Z-axis.
translateMX	Determines if the model can be moved along the X-axis.
translateMY	Determines if the model can be moved along the Y-axis.
translateMZ	Determines if the model can be moved along the Z-axis.
showShadows	Leave <code>true</code> - Don't change
autoSceneUnderstanding	Leave <code>false</code> - Don't change
autoExpSceneStart	Leave <code>false</code> - Don't change
useSceneMesh	Leave <code>false</code> - Don't change
visualizeSceneMesh	Leave <code>false</code> - Don't change
trackFloor	Leave <code>false</code> - Don't change
useSceneObserving	Leave <code>false</code> - Don't change

# Learn »Experimental State and Research Software«!

## Experimental State and Research Software

Since MARBLE is a research software, it is not 100% stable and reliable. Many functions may not work directly as expected. Try everything at your pace and take your time.

# Learn »Quick Fixes«!

## Quick Fixes

### Creator Environment

#### Back Button does not react

It might occur that you hit the "Back" button but nothing happens. The cause is often Unity's behaviour when exporting or building packages or apps. The static Creator loses the window instance and can not execute commands on it.

1. close the Creator Window (right click on the tab and click "close tab").
2. if step 1. does not work, restart Unity.

### Configurations

#### Malfunction at first start

If the app does not apply any configs on the first start, the app might not have initialized its configuration data correctly. MARBLE uses some space in the picture library to store data like configs. On the first start, it might happen that it does not get access to write all the data to disk before starting the experience. Steps to fix: make sure that:

1. the directory structure is created correctly --> on the device manager, head to "File Explorer > User Folders > Pictures". There has to be the following path: MARBLE > Data > Configs.
2. there are seven config files (config, debugconfig, featureConfig, menuConfig, netconfig, pointerconfig, studyconfig).
3. these config files are not empty (0 bytes).

If one of those issues occur do the following:

1. create the folders stated in step 1.
2. upload all files manually. Therefore, chose the "Upload" button, head to the Unity project folder "Assets/\_MARBLECreator/Data/Configs" and upload the configs one by one.
3. make sure that the uploaded configs have the right names (config, debugconfig, featureConfig, menuConfig, netconfig, pointerconfig, studyconfig).
4. restart the MARBLE app.