

# 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.

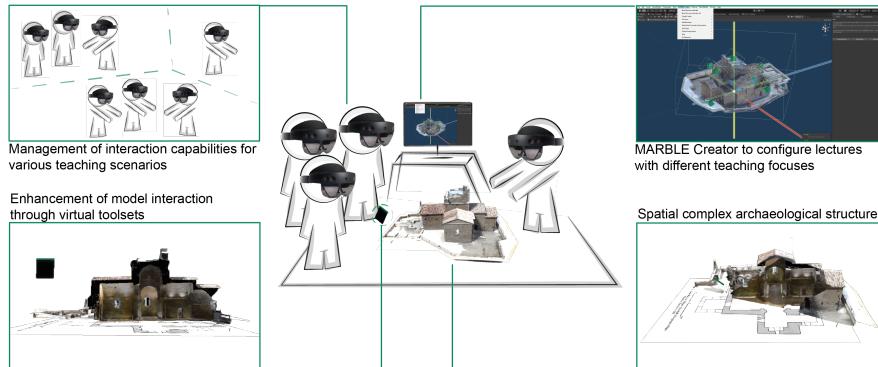
### MARBLE Background

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.

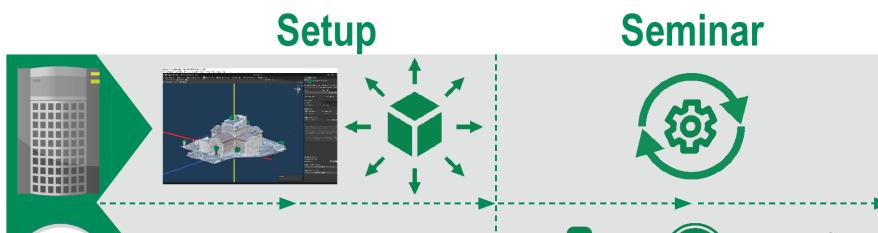
# Learn »A Mixed Reality Approach to Enhance Archaeological Higher Education«!

## 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 »MARBLE Videos«!

## MARBLE Videos

We recorded some videos to show the basic functionality of MARBLE and what can be done with it. The videos are reachable at this [link](#)  directing to YouTube.

At the lower right of the video another view of the scene is displayed showing the real-time recording of a second HoloLens2 connected to the MARBLE Creator. It shows that MARBLE displays the model for every user at the same place with the same orientation and size.



## Interested? What's Next?

- Read the IEEE ISMAR Paper 
- Watch the videos 
- The MARBLE Concept
- Get Access
- Get Started

# 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\_1.0.1« which you have to [request from the author](#). »marble\_isengard\_1.0.1« is the slim version of »marble\_mordor\_1.0.1«. 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:

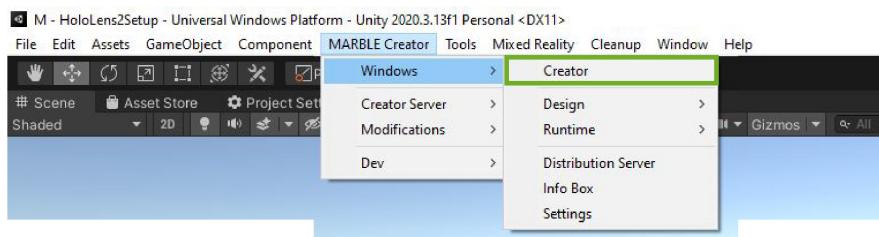
# Learn »Getting Started«!

## Getting Started

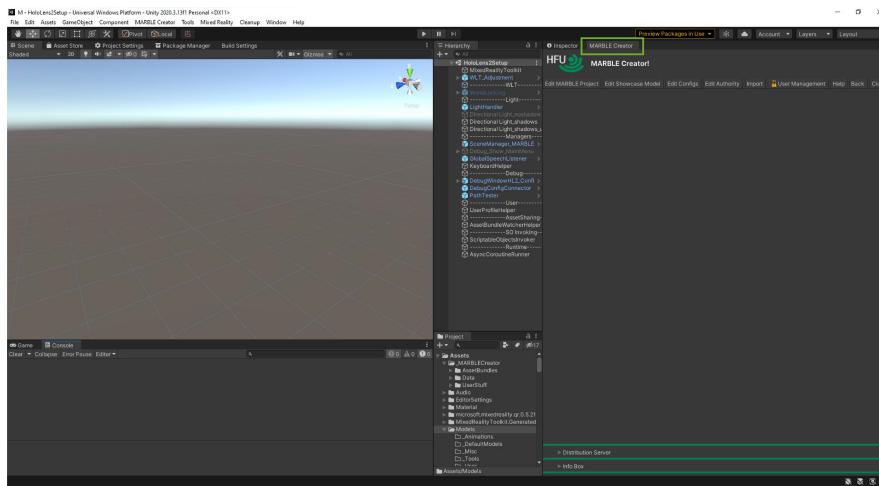
First of all make sure that you have installed MARBLE as stated in [Install & Setup](#).

## 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.

## Runtime Interactions

## 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.

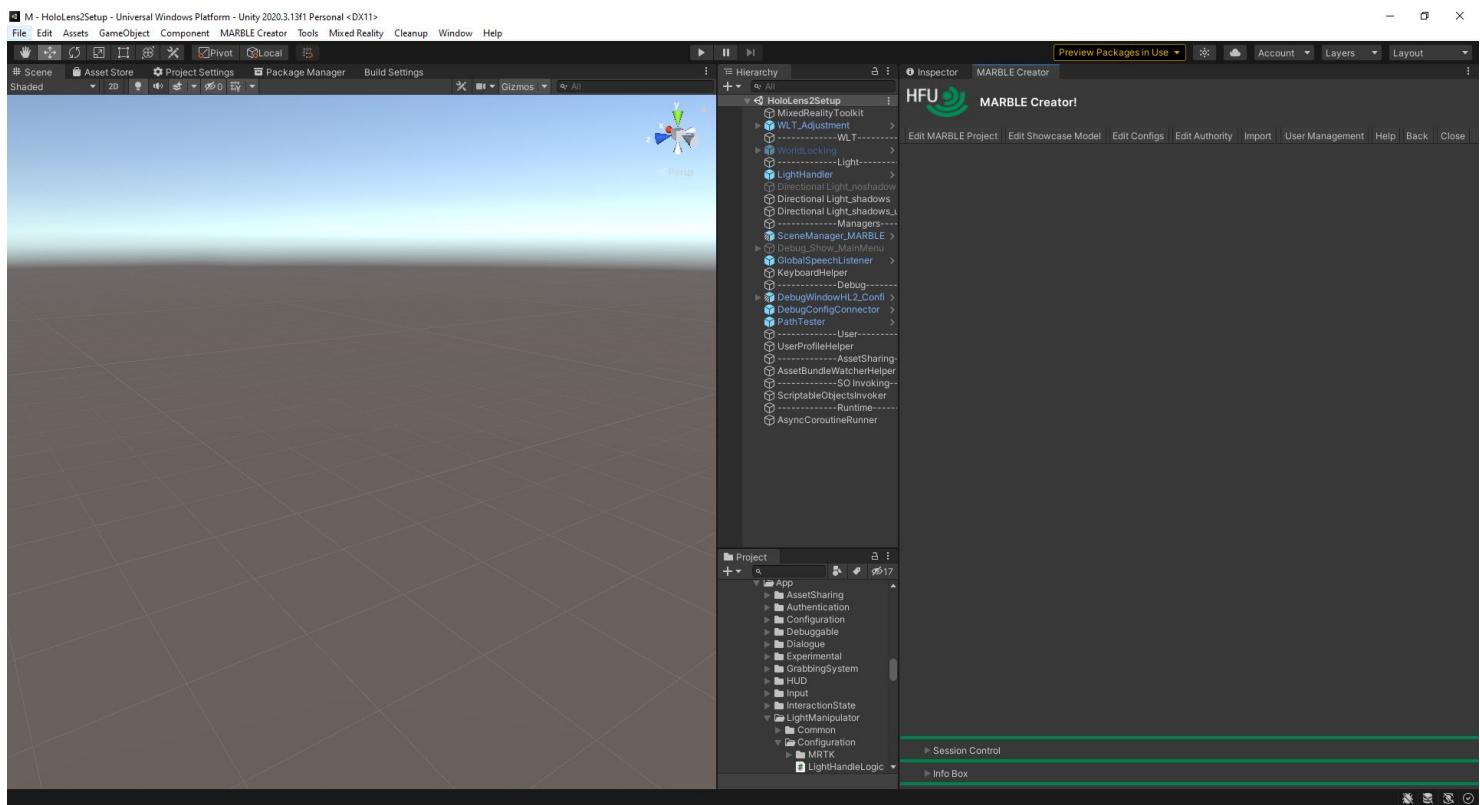
## Asset Bundles and Updates

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".

# Learn »Unity and Creator Overview«!

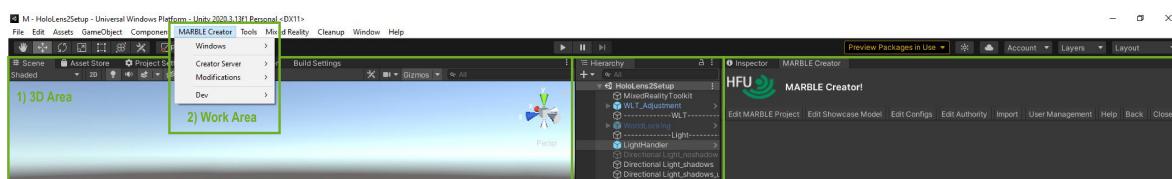
## Unity and Creator Overview

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.

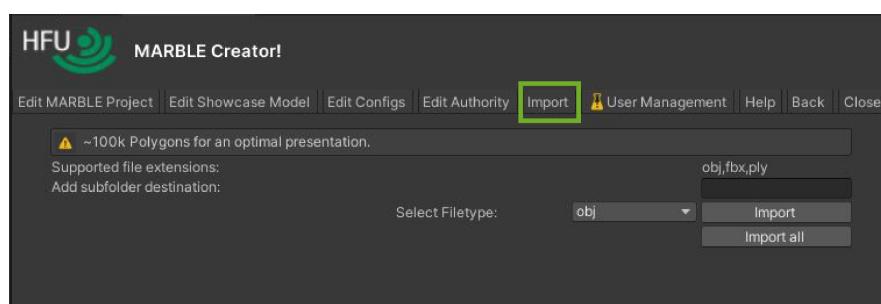
### Navigate to import

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.

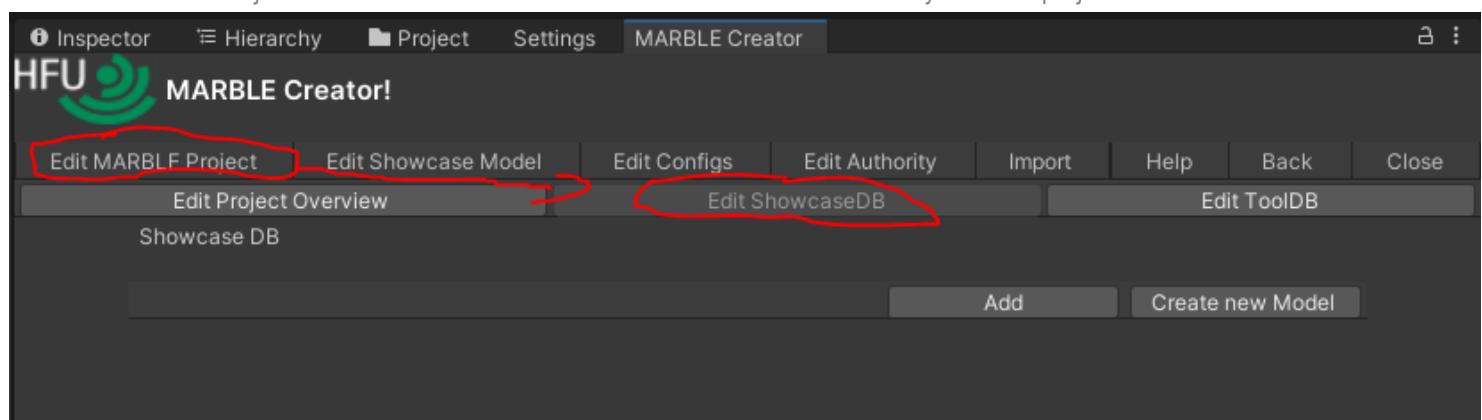
# Learn »Showcase Databases«!

## 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.

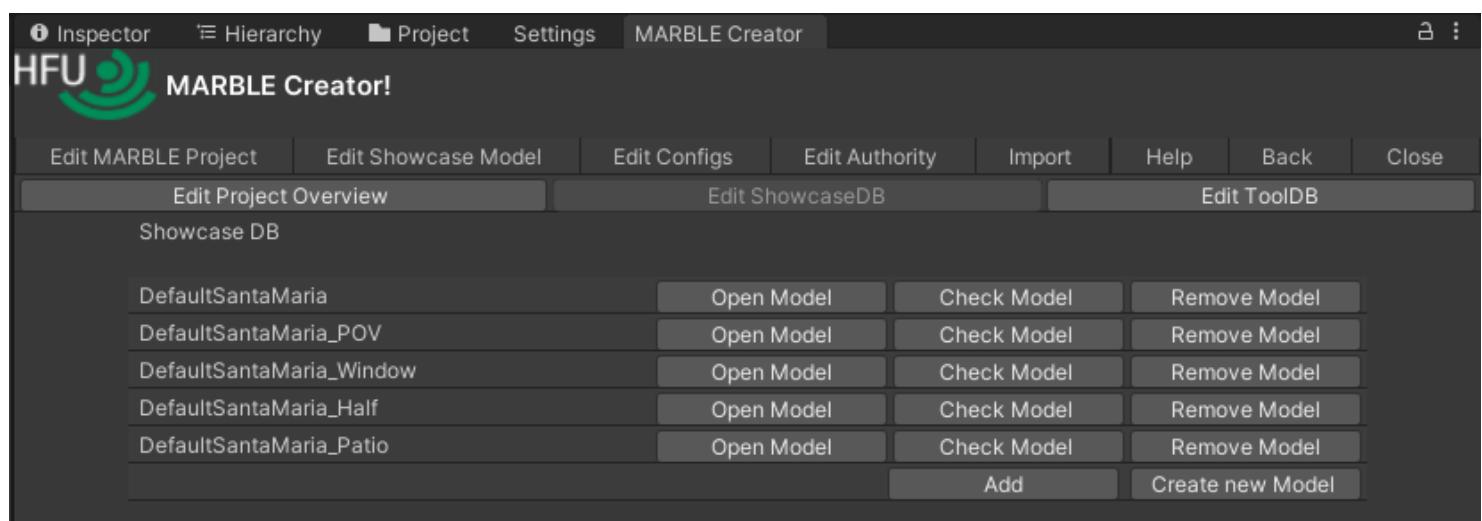
## Navigation

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 menu bar with 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

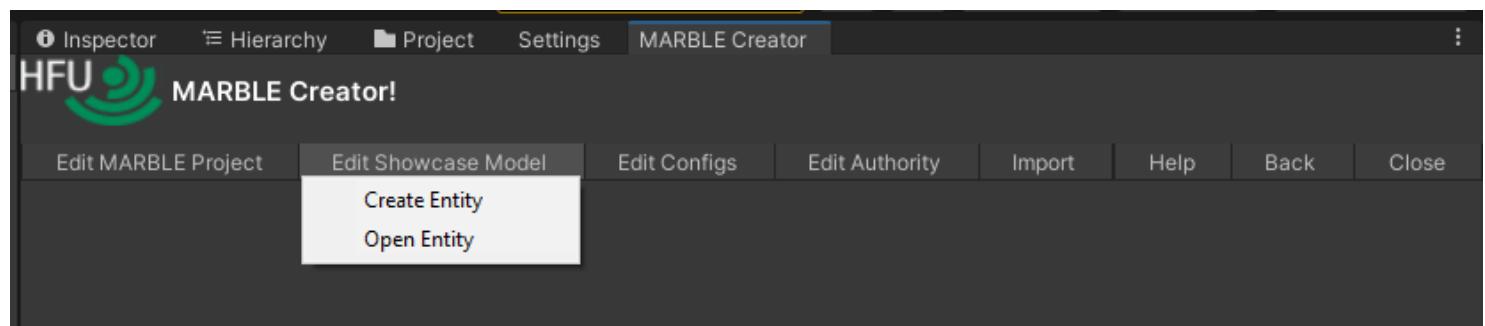
# Learn »Showcase Models«!

## 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.

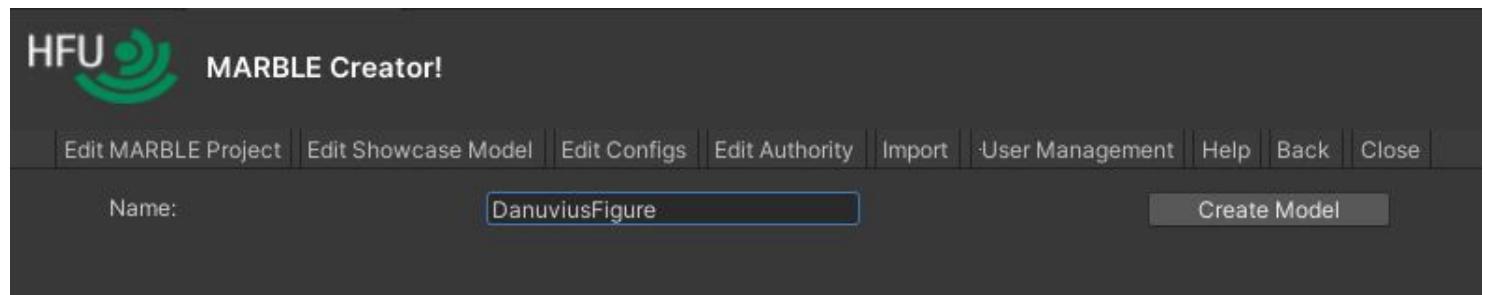
## Navigation

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

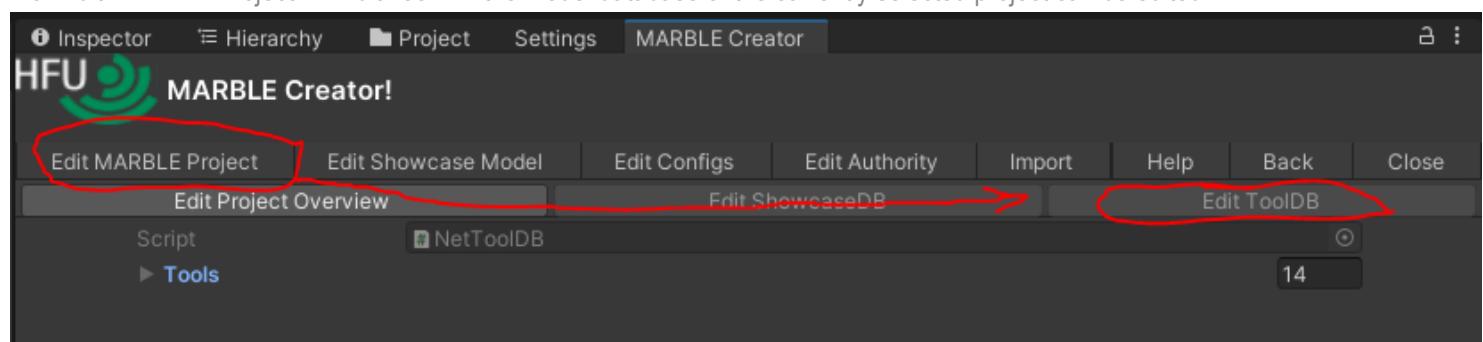
# Learn »Tool Databases«!

## 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.

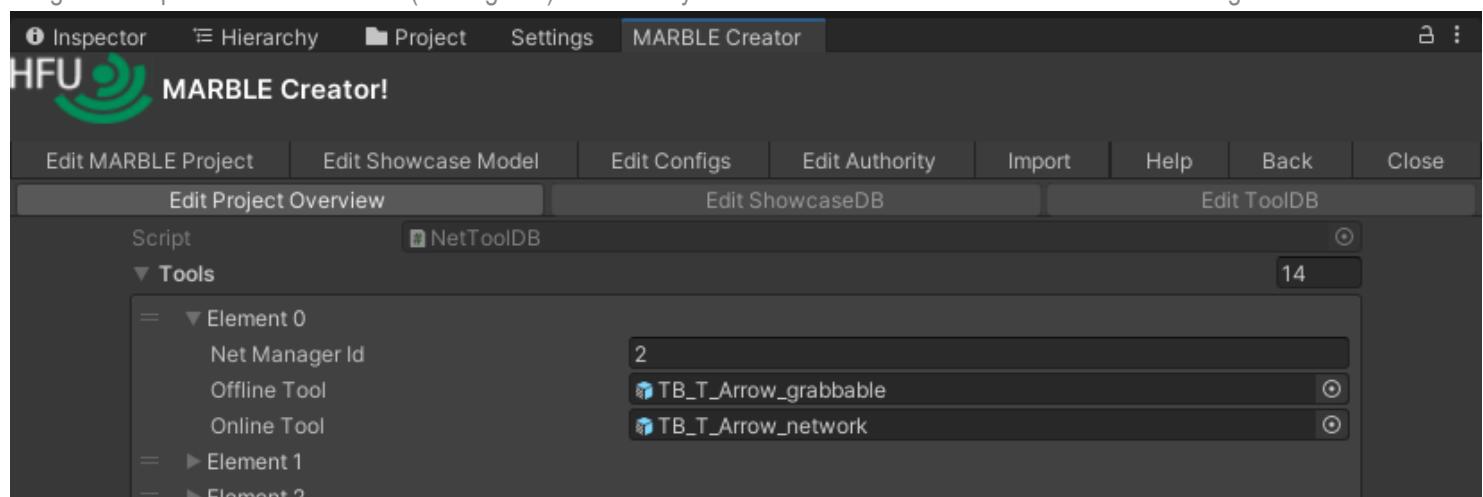
## Navigation

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



## Add & Remove Tools

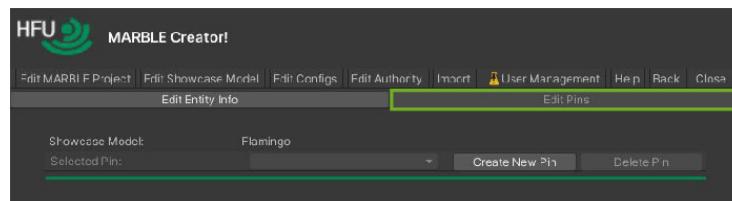
Currently, adding tools is implemented, but not supported in a user-friendly manner. In order to add a new tool, you have to create a new tool by deriving a new prefab from the "TB\_Tool\_grabbable\_network" and one from "TB\_Tool\_grabbable". In prefab mode, you have to add the desired model and functionality 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 »Pins, Details & Connections«!

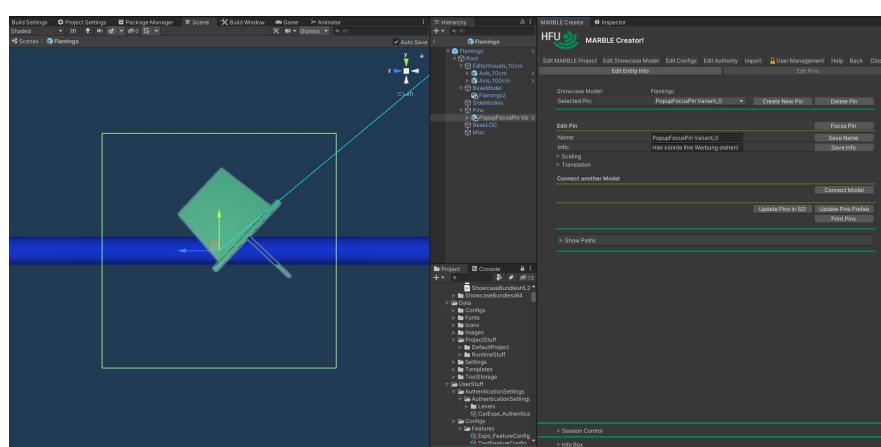
## Pins, Details & Connections

With "pins", one can add relevant information to a model or connect another model to be accessed from this model. With pins, the Creator offers the possibility to design an overview-focus kind of hierarchy where one model serves as low resolution overview for navigation and several high resolution detail models as actual learning material. This is shown in [this video](#). Pins are part of a model and only accessible through "Edit Showcase Model > Select Model > Open Entity > Edit Pins".



## Create Pins

Simply create a pin by clicking on the "Create New Pin" button at the "Edit Pins" panel of an opened model. By doing so, the Creator spawns a 3D pin in the model itself, seen on the left side of the image below. The Creator also allows the user to modify the pins metadata.



## Pin Placement

To properly place a pin in the model, one can either use the experimental scaling and translation dropdowns beneath the metadata in the "Edit Pin" section or use Unity's scene view. In the scene view, one can place a pin the same way a [model is placed](#).

## Connect other Models

Pins can also contain a connection or reference to another model which, by this, can be accessed through the original model. To connect a model, simply navigate to the button "Connect Model" in the "Connect another Model" section.

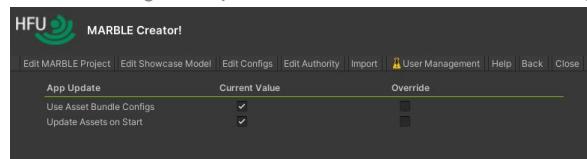
# Learn »Configs & Effects«!

## Configs & Effects

MARBLE stores most of its settings in configuration files. These files are not only necessary for the app itself, but also enable the configuration of functions for sessions, depending on their didactic goal. For example, one can remove all interactions from all clients to turn the MARBLE Player into a simple, one-way presentation tool. In combination with the authentication and authority mechanisms, one can configure a session so that only one or two people (e.g. the professor or tutor) can control the session and the models. If you would like to learn more about this, read [Concept & Architecture](#).

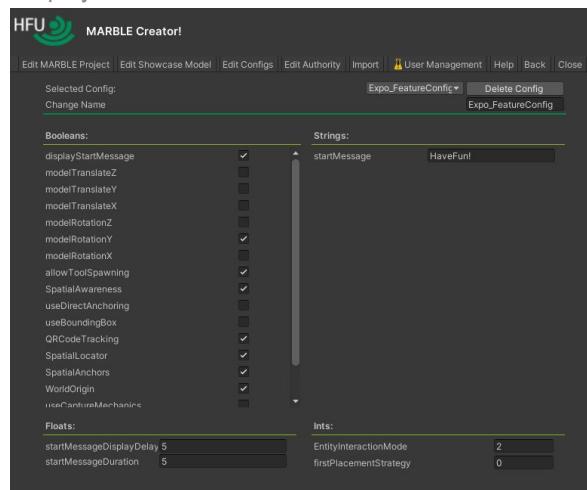
## Quick Config Edit

The "Quick Config Edit" panel allows the user to quickly configure certain settings that overwrite the settings from the configuration files when exporting a MARBLE project. The "Quick Config Edit" panel is located under "Edit Configs > Quick Config Edit".



## Session Configs

In the "Edit Session Config" area, the user can configure the session for the MARBLE project. The user can configure features such as menu items or define whether the model can be moved on a specific axis or not. In the "Selected Config:" drop-down menu, the user can select which configuration they want to change or delete. The four types of configs available are Boolean, String, Integer or Float. All four configurations to be changed are displayed in the main area of the "Edit Session Configs" panel.



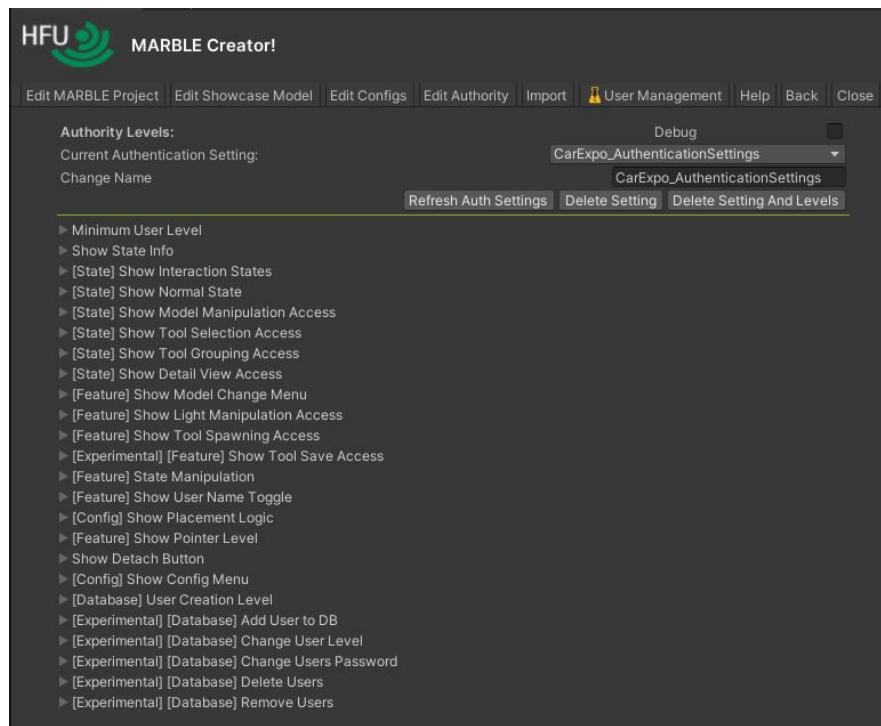
## Configs List

# Learn »Authority & Rights«!

## Authority & Rights

MARBLE is based on an authority and right logic to enable fine-grained access to features and information. Every supported feature has its own level which a user needs to gain access to it. For more information, read [Layers of Configuration](#). The editing of the authorities can be accessed through "Edit Authority".

The panel displays the current authority rules, which can be changed with the drop-down at "Current Authority Settings:" (note: the name can be misleading. Sometimes the Creator calls it "Authentication". This is an old remain of a different system).



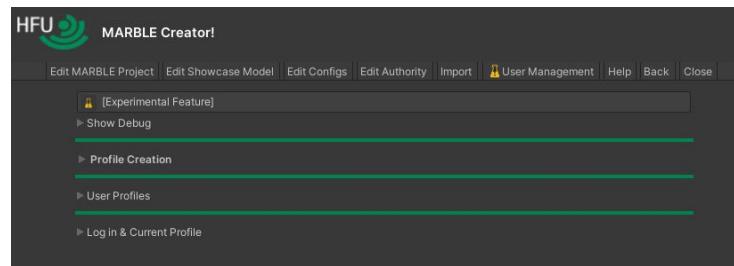
Beneath the general settings, the Creator lists all rules for the different features. They all consist of a display name, a description and a level. The lower the user's level is, the more features can be accessed by him/her (0 = highest access, 99 = lowest access). For example, if one changes the level from the "Show Normal State" to 0, all users with a level higher than 0 (1 and above) will not be able to see the normal state in their runtime state menu.

 There are a few special entries in the list of authority levels. One should not change any entry with the "[Database]" tag. Also, the "Minimum User Level" is a special setting. The "Minimum User Level" is meant to bypass the actual user level. Setting the "Minimum User Level" to a low value at design or runtime will give the user access to otherwise hidden features. During runtime, one can quickly give the users access to several features without bothering about important changes.

# Learn »User Management«!

## User Management

MARBLE implements a really simple user management. The core features are creating and accessing user profiles (like accounts) to store information like name, main color and authority levels/ rights.

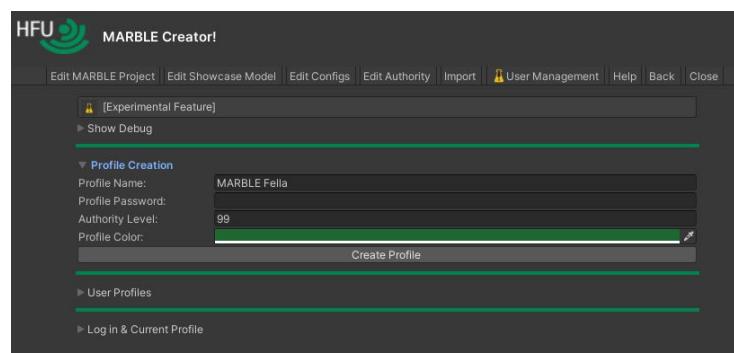


## Profile System

The profile system is in an experimental state. It is only meant to offer the possibility to assign features and visibilities to clients at runtime. For example, you can create a tutor account in which the tutor can log in at runtime to be able to move the model without having the need to also give this access to all other students. Nonetheless, MARBLE can be run without any additional profiles as already created. Every client which connects will get a temporary profile while being part of the experience. After they disconnect, the server deletes their data. There are two profiles which should not be touched by any means. The first one is the admin profile, which is the root profile of the server. No other profile should have a lower or the same level as the admin. The second is the default "Student" profile. This profile serves as a blueprint for all temporary connected clients without own profile.

## Create Profile

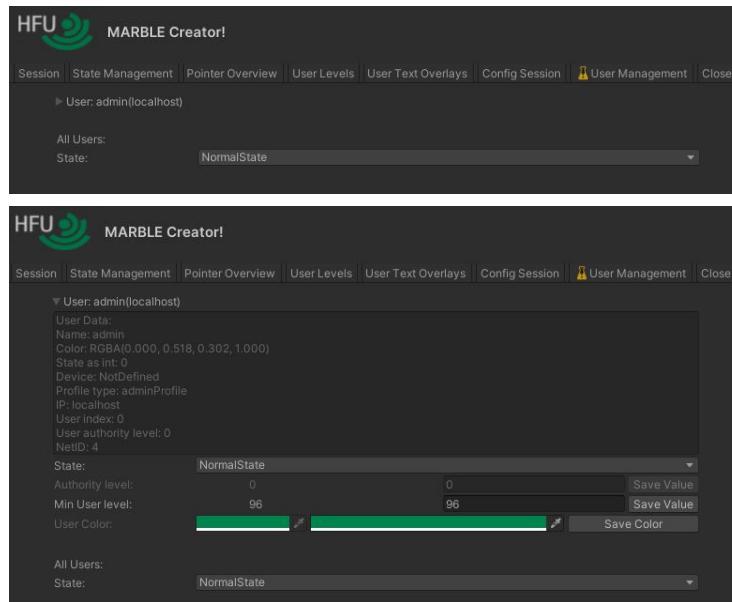
One can create a profile in the "Profile Creation" fold-out section. Enter name, password, level and color and click "Create Profile".



## User Profiles

## Session Management

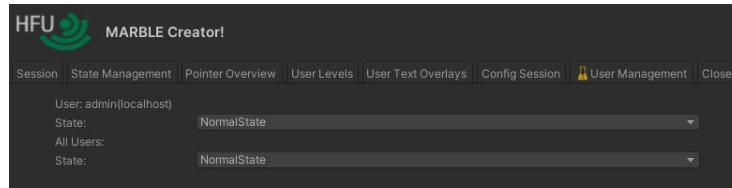
The session management panel gives one a quick overview of the current connection and showcase state. Under "Session > User Overview" one can find all connected clients, quick review their data and change their **interaction state**, **authority level** and color. This panel also gives the possibility to change the **interaction state** of all users at once.



Under "Session > Showcase" once can control the displayed showcase. The fold-out "Spawnables" displays a list of all models of the session. "IsDetail" indicates whether the model is accessed from another model or not. Simple click on "Show this" next to a model to display it on the clients. One can also use the "Previous" and "Next" controls to switch between models.

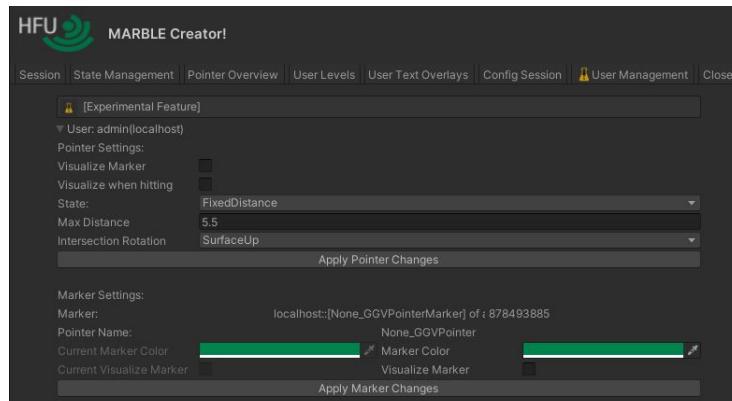
## State Management

The state management displays all users and their **interaction state**.



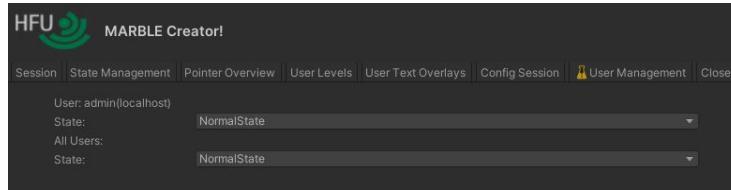
## Pointer Overview

In the "Pointer Overview" panel one can manage the synchronized pointers and their visibility.



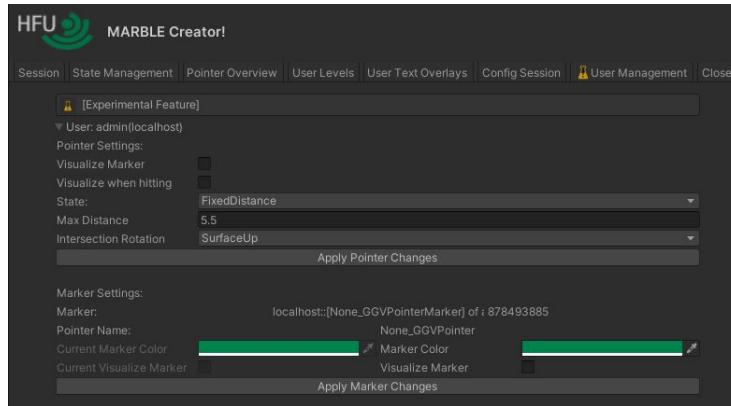
Every change has to be applied by clicking either "Apply Pointer Changes" or "Apply Marker Changes". In the following each setting is explained.

The State Management displays all users and their **Interaction State**.



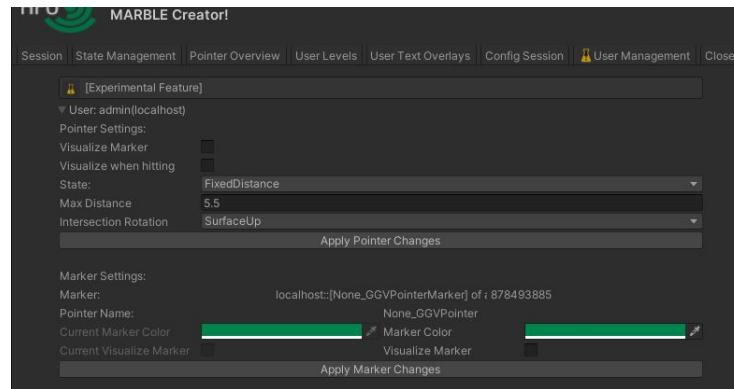
## Pointer Overview

In the "Pointer Overview" panel one can manage the synchronized pointers and their visibility.



Every change has to be applied by clicking either "Apply Pointer Changes" or "Apply Marker Changes". In the following each setting is explained.

IDENTIFIER	DESCRIPTION
<b>Pointer Settings</b>	
Visualize Marker	Determines if the pointers should be displayed or not.
Visualize when hitting	Determines if the pointers should be visualized even when they do not hit the model.
State	When "Visualize when hitting" is enabled one can choose between "LastPosition" or "FixedDistance" to determine how the pointer marker will be displayed when not hitting the model.
Max Distance	When "FixedDistance" is selected "Max Distance" determines how far the pointer marker will be displayed.
Intersection Rotation	How the marker is displayed on the model. "AsCursor" is recommended as this orientates the marker like the not synced MRTK cursor.
<b>Marker Settings:</b>	
Marker	Name of the marker the following settings belong to.
Pointer Name	The name the marker belongs to. Probably either left or right hand.
Current Marker Color	The colour of the marker. The colour can be independently changed for every marker (left or right hand). The colour will be reset when the marker disappears once. To change the colour persistently, change the profile colour of the user.



Every change has to be applied by clicking either "Apply Pointer Changes" or "Apply Marker Changes". In the following each setting is explained.

IDENTIFIER	DESCRIPTION
<b>Pointer Settings</b>	
Visualize Marker	Determines if the pointers should be displayed or not.
Visualize when hitting	Determines if the pointers should be visualized even when they do not hit the model.
State	When "Visualize when hitting" is enabled one can choose between "LastPosition" or "FixedDistance" to determine how the pointer marker will be displayed when not hitting the model.
Max Distance	When "FixedDistance" is selected "Max Distance" determines how far the pointer marker will be displayed.
Intersection Rotation	How the marker is displayed on the model. "AsCursor" is recommended as this orientates the marker like the not synced MRTK cursor.
<b>Marker Settings:</b>	
Marker	Name of the marker the following settings belong to.
Pointer Name	The name the marker belongs to. Probably either left or right hand.
Current Marker Color	The colour of the marker. The colour can be independently changed for every marker (left or right hand). The colour will be reset when the marker disappears once. To change the colour persistently, change the profile colour of the user.
Current Visualize Marker	If the marker should be visualized. In case "Visualize Marker" of the "Pointer Settings" section is disabled, the settings of "Current Visualize Marker" will be overwritten.

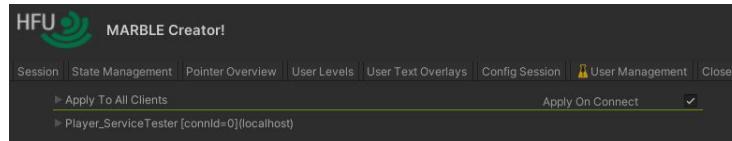
⚠ The bug that some changes under "Marker Settings" are not applied is known. The current fix is simply to make the change again.

## User Levels

⚠ The bug that some changes under "Marker Settings" are not applied is known. The current fix is simply to make the change again.

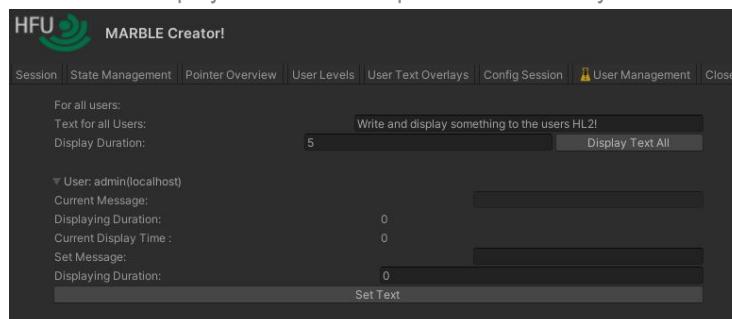
## User Levels

The "User Levels" panel is a runtime version of the [Authority & Rights](#) panel. At the top one can change the authority settings for all users at once.



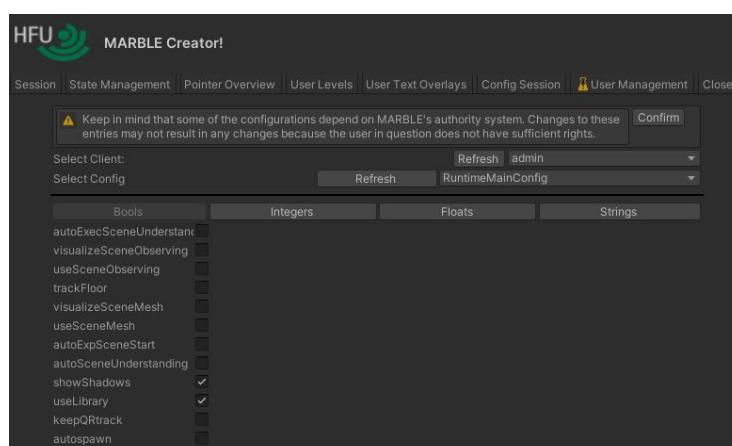
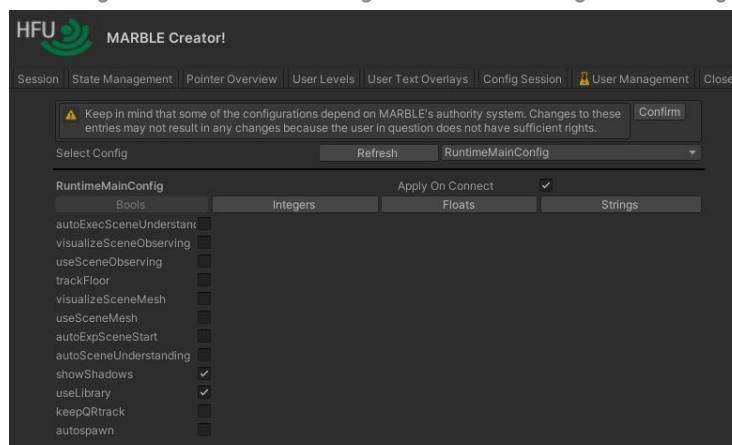
## User Text Overlays

At this panel one can display small messages on the clients. The messages can be displayed at one or all connected clients. It is also possible to enter a duration the text should be displayed before it disappears automatically.



## Config Session

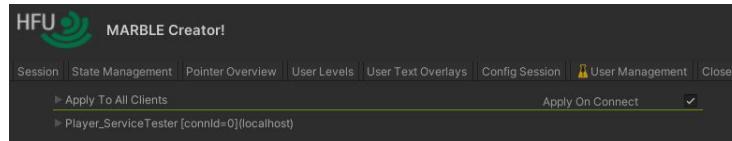
The "Config Session" panel is a runtime version of the [Session Configs](#) panel. Under "Config Session > General Config" the configs for all connected clients can be edited. "Config Session > Client Configs" enables a editing of the configs per client.



⚠ The bug that some changes under "Marker Settings" are not applied is known. The current fix is simply to make the change again.

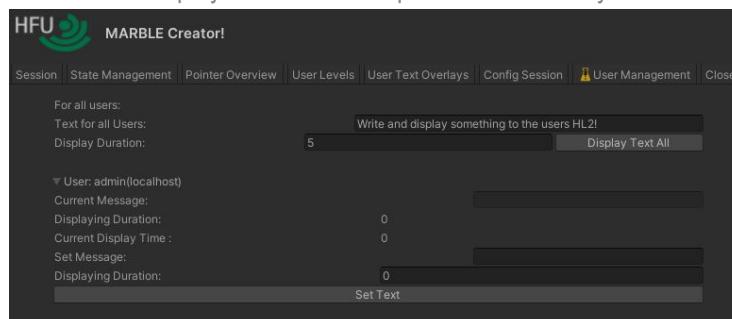
## User Levels

The "User Levels" panel is a runtime version of the [Authority & Rights](#) panel. At the top one can change the authority settings for all users at once.



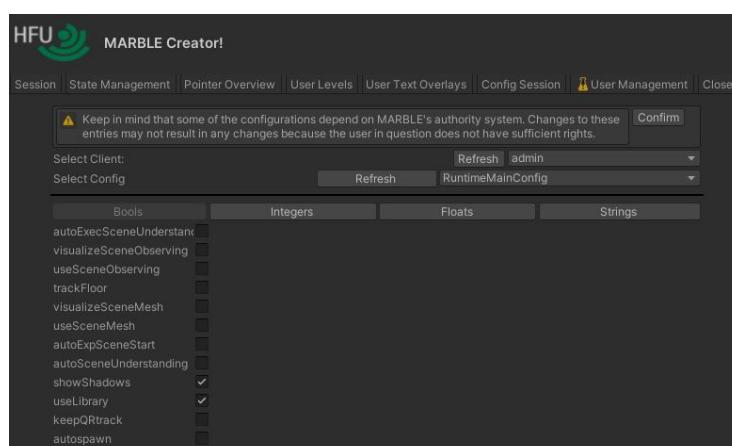
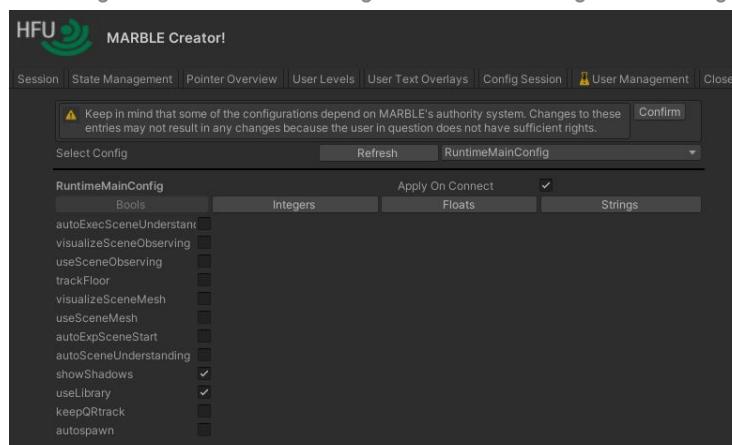
## User Text Overlays

At this panel one can display small messages on the clients. The messages can be displayed at one or all connected clients. It is also possible to enter a duration the text should be displayed before it disappears automatically.



## Config Session

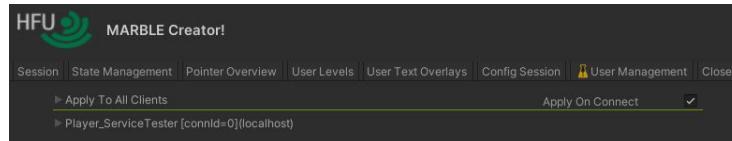
The "Config Session" panel is a runtime version of the [Session Configs](#) panel. Under "Config Session > General Config" the configs for all connected clients can be edited. "Config Session > Client Configs" enables a editing of the configs per client.



⚠ The bug that some changes under "Marker Settings" are not applied is known. The current fix is simply to make the change again.

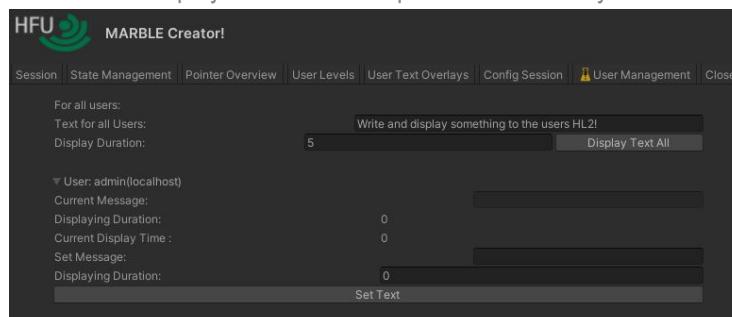
## User Levels

The "User Levels" panel is a runtime version of the [Authority & Rights](#) panel. At the top one can change the authority settings for all users at once.



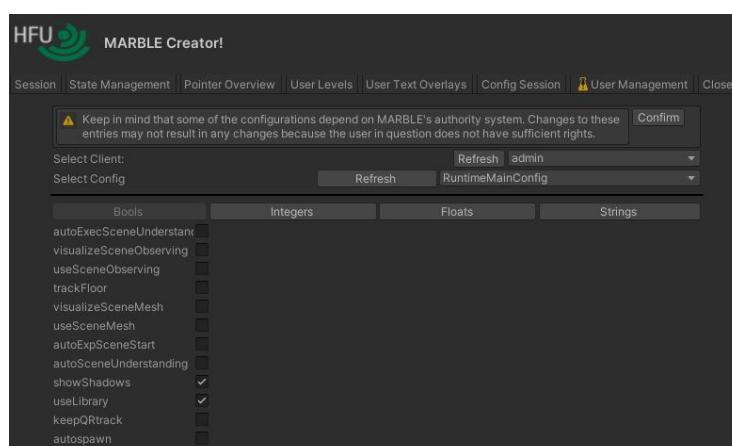
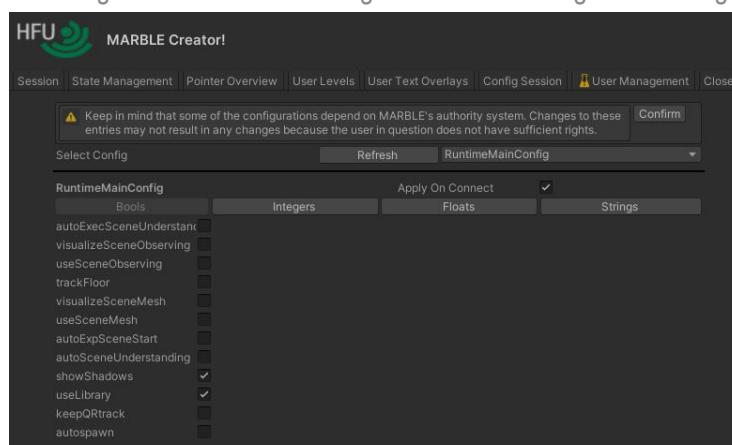
## User Text Overlays

At this panel one can display small messages on the clients. The messages can be displayed at one or all connected clients. It is also possible to enter a duration the text should be displayed before it disappears automatically.



## Config Session

The "Config Session" panel is a runtime version of the [Session Configs](#) panel. Under "Config Session > General Config" the configs for all connected clients can be edited. "Config Session > Client Configs" enables a editing of the configs per client.



# 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.

### Import Models Without Textures

See [Import Models > Models Without Textures](#).

## Pins

### Metadata Will Not Be Applied

See [Import Models > Description of Pin will not apply](#).

### Empty Pin Descriptions at Runtime

See [Import Models > Empty Pin Descriptions at Runtime](#).

## Player Environment

### Non-Executing Menus

It can happen that you press something in a holographic menu but nothing happens. This often happens if the connection gets messed up. Make sure that the connection is established and reopen the affected menu.

# Learn »Known Errors«!

## Known Errors

As MARBLE is a research software, some errors may still occur. The following is a list of known errors that can be ignored.

! [11:01:10] SerializedObjectNotCreatableException: Object at index 0 is null  
UnityEditor.Editor.CreateSerializedObject () (at <01e901910c5242498b7e69f6d2d8d500>:0)

! [11:28:43] No Prefab open. Abort "UpdatePrefab()"  
UnityEngine.Debug:LogError (object)

! [13:56:42] System.IO.DirectoryNotFoundException: Could not find a part of the path 'C:\Users\Marc\Git\MARBL' at System.IO.\_Error.WinIOError (System.Int32 errorCode, System.String maybeFullPath) [0x000f7] in <695d1c>

! [14:01:54] The asset at Assets/\_MARBLECreator/Data/Fonts/LTe50148.ttf has been scheduled for reimport due to a change in its file path.  
Doing this can lead to the AssetDatabase returning two versions of the same asset.

! [14:03:12] The asset at Assets/\_MARBLECreator/Data/Fonts/LTe50148.ttf has been scheduled for reimport due to a change in its file path.  
Doing this can lead to the AssetDatabase returning two versions of the same asset.

! [14:11:20] InstanceID: -283524  
AssetPath:

! [14:11:20] [FileUtils] Strings for path and/ or filename are empty. Caller: ""Abort!  
UnityEngine.Debug:LogError (object)