Sheath Script 1.1

Game Engine: Unity

Kevin Iglesias - www.keviniglesias.com

Contact Support:
support@keviniglesias.com

Online Documentation

CHANGELOG

Version 1.1

- Added new animations and sheath examples (Katana and Leg Knife).
- Added an intuitive custom inspector.
- Added easier and better compatibility with Animation Events.
- Now you can create infinity number of sheaths.

FOLDER CONTENTS

Animations.

A total of 38 animations compatible with Mecanim for sheathe/unsheathe in different sheath locations (hips, back, lower back, right, left, both hands…).

Prefabs.

Example of uses. These assets are prepared with the scripts already configured for taking as reference for your game characters.

Scripts.

Two scripts are included, the main script is needed to be added as a component to your game character. The other is optional and can be used in the animation state of your Animator Controller. Further details and configuration are explained below.

Source Models

The FBX models and the textures used by the example character and the weapons.

Demo scene

A demo scene containing all the prefabs separated in two groups: Single hand sheaths and both hand sheaths.

Documentation

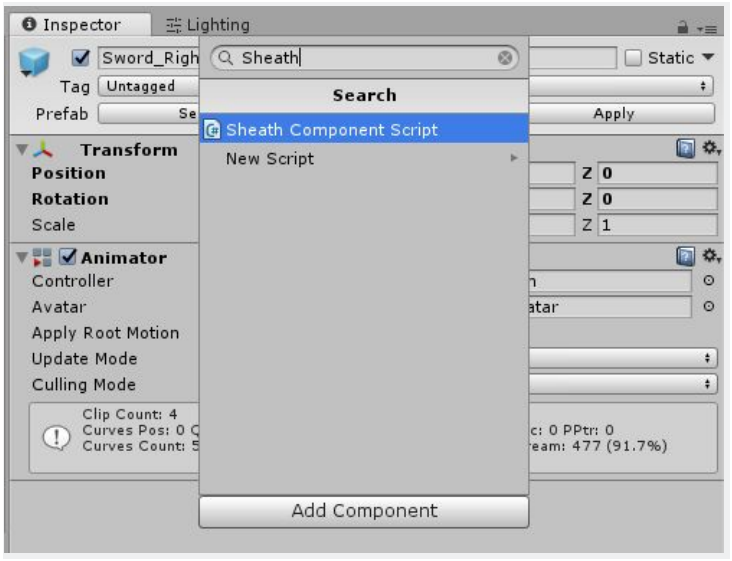
This documentation in PDF file format.

HOW TO USE

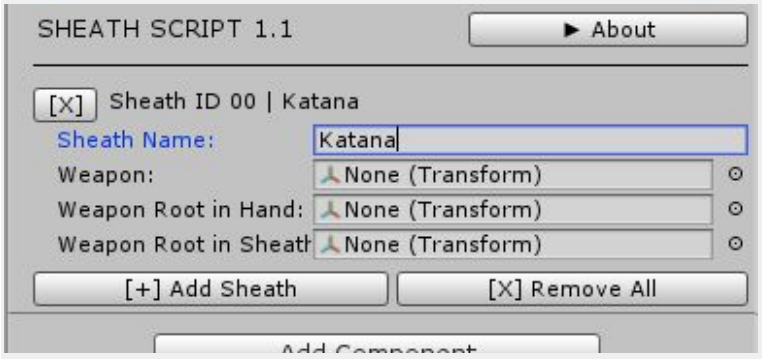
First of all, you need to add the main script to your game character: the SheathComponentScript (MonoBehaviour component).

SheathComponentScript (Main Script)

The SheathComponentScript must be added as a component to your game character where the Animator component is.



Adding the SheathComponentScript from the Inspector.



Adding one sheath and renaming it for easy identify.

Add at least 1 sheath. (You only need 1 per weapon, you can either sheathe and unsheathe using only one Sheath Entry)

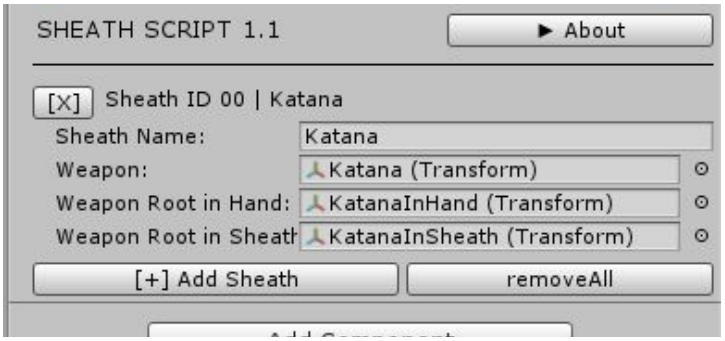
Sheaths fields

- **Sheath Name.** Assign a name to the sheath. Useful for identifying right from left sheaths in case you have more than one.

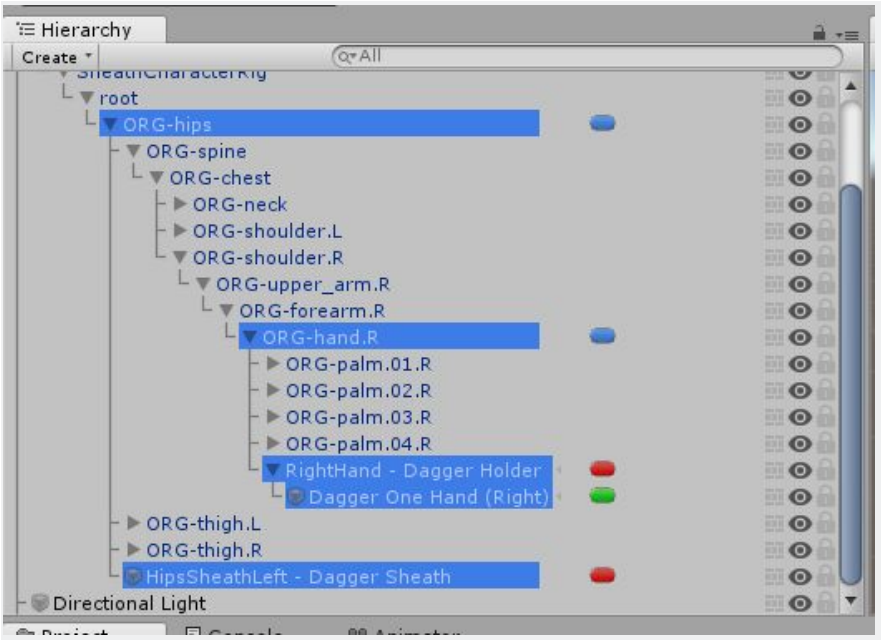
- **Weapon.** This field is for the weapon GameObject that will be sheathed and unsheathed. Drag here your weapon.
  It is **important** that its position and rotation remain to 0 at all of the time. For fitting the weapon in the sheath or hand create a new Transform and use it as root for the weapon. Move and rotate the root but keep the weapon Transform position and rotation at 0.

- **Weapon Root In Hand.** This will be the parent of the weapon when it is unsheathed, needs to be the children of the hand bone of your character. One Weapon Root In Hand can have as many weapons as children as you wish.
  You can move and rotate this GameObject so it fits well when your character holds the weapon.
  You need to create this GameObject and parent it to the hand bone of your character.

- **Weapon Root In Sheath.** This is the position and rotation where the weapon will stay when it is sheathed. One Weapon Root In Sheath can have as many weapons as children as you wish.
  You can move and rotate this GameObject so it fits well when the weapon is sheathed.
  You need to create this GameObject and parent it to a bone of character. If the sheath is at the hips, parent it to the hips bone but if the sheath is at the back you should parent this to the chest bone.



Example of a good configuration of the SheathComponentScript



Hierarchy of the example configuration.

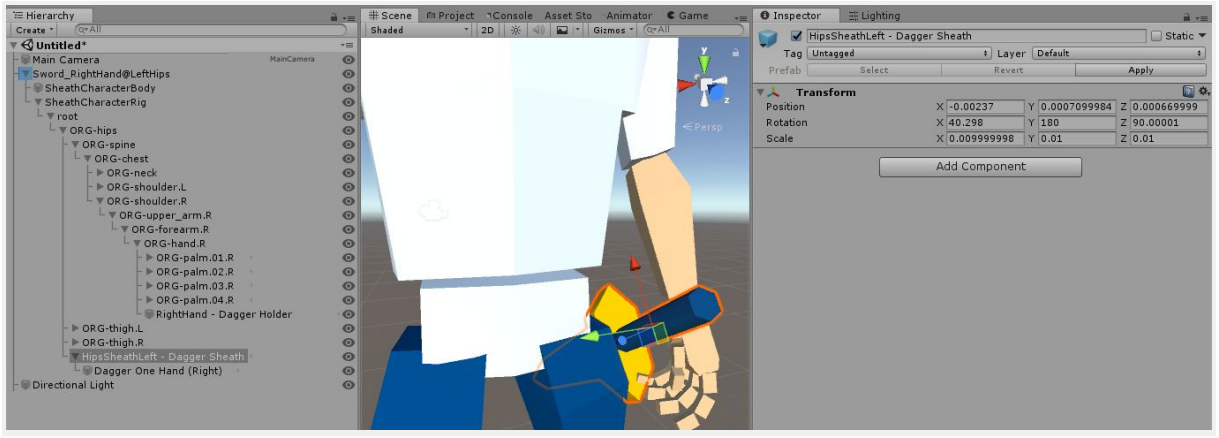
The blue dots are the bones of your game character. In this case the GameObject Hips ('ORG-hips') is the parent of the sheath (Sheath position field) and the Right Hand ('ORG-hand.R') is the parent of the weapon holder (Hand position field).

The red dots are the GameObjects you need to create and parent to the correct bone.

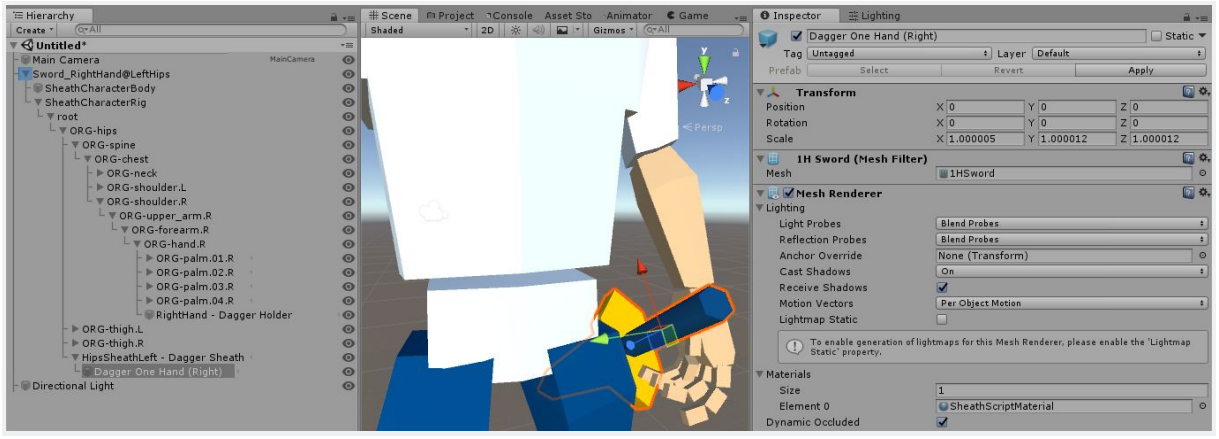
In this case the weapon holder 'RightHand - Dagger Holder' is the Weapon Root In Hand field. You can move and rotate this GameObject so it fits well when the weapon is unsheathed.

The 'HipsSheathLeft - Dagger Sheath' is the Weapon Root In Sheath field in the script. You can move and rotate this GameObject so it fits well when the weapon is sheathed.

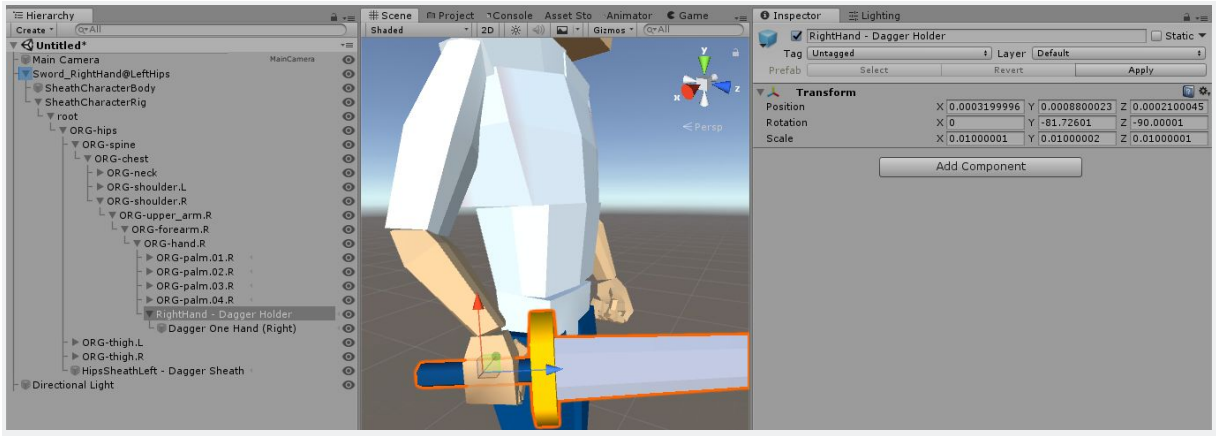
The green dot is the weapon to sheath and unsheath (Weapon field). This GameObject must remain at position (0,0,0) and rotation (0,0,0).



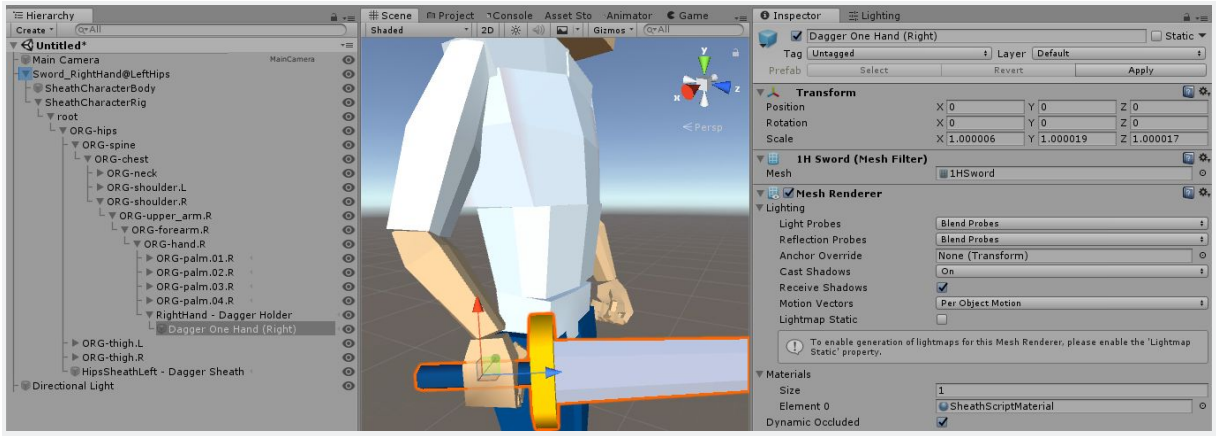
Sheath position and rotation adjusted to fit in hips



Weapon sheathed remains at position (0,0,0) and rotation (0,0,0)



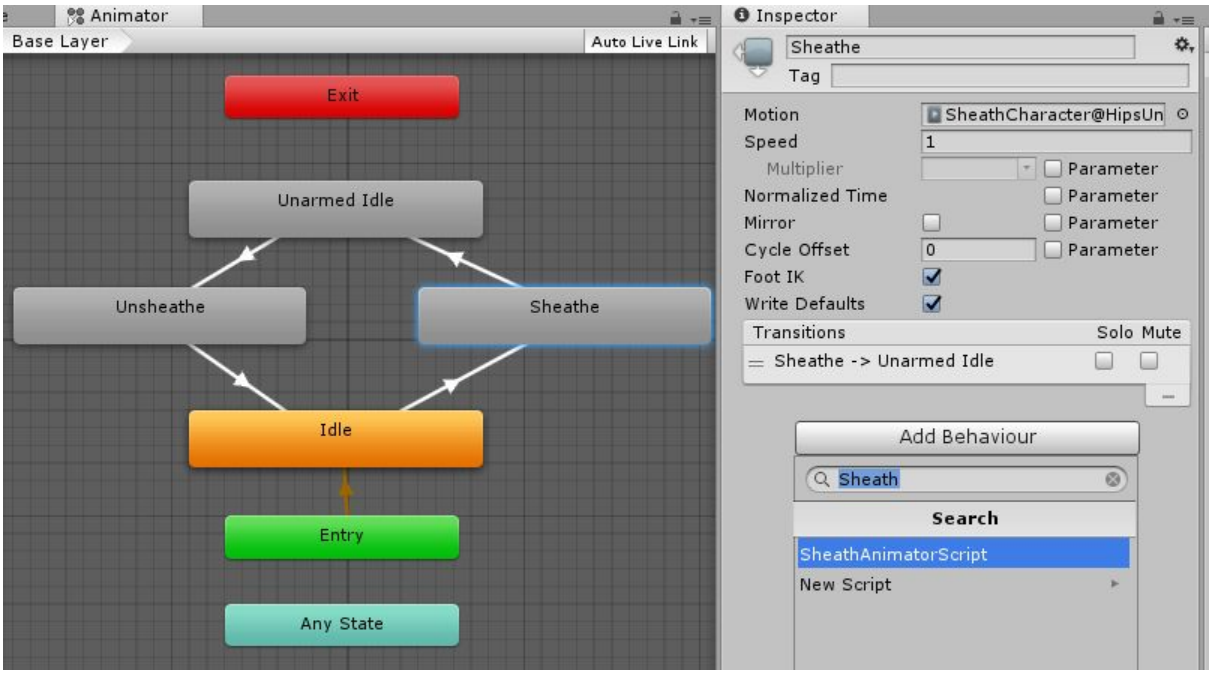
Hand position and rotation adjusted to fit the weapon in hand



Weapon unsheathed remains at position (0,0,0) and rotation (0,0,0)

Sheathe and Unsheath using StateMachineBehaviour

Using the **SheathAnimatorScript** you can call sheathe or unsheathes. Just add the script as a StateMachineBehaviour to an animator state with the sheathe or unsheathe animation.



Add the SheathAnimatorScript in an Animator State from the Inspector.

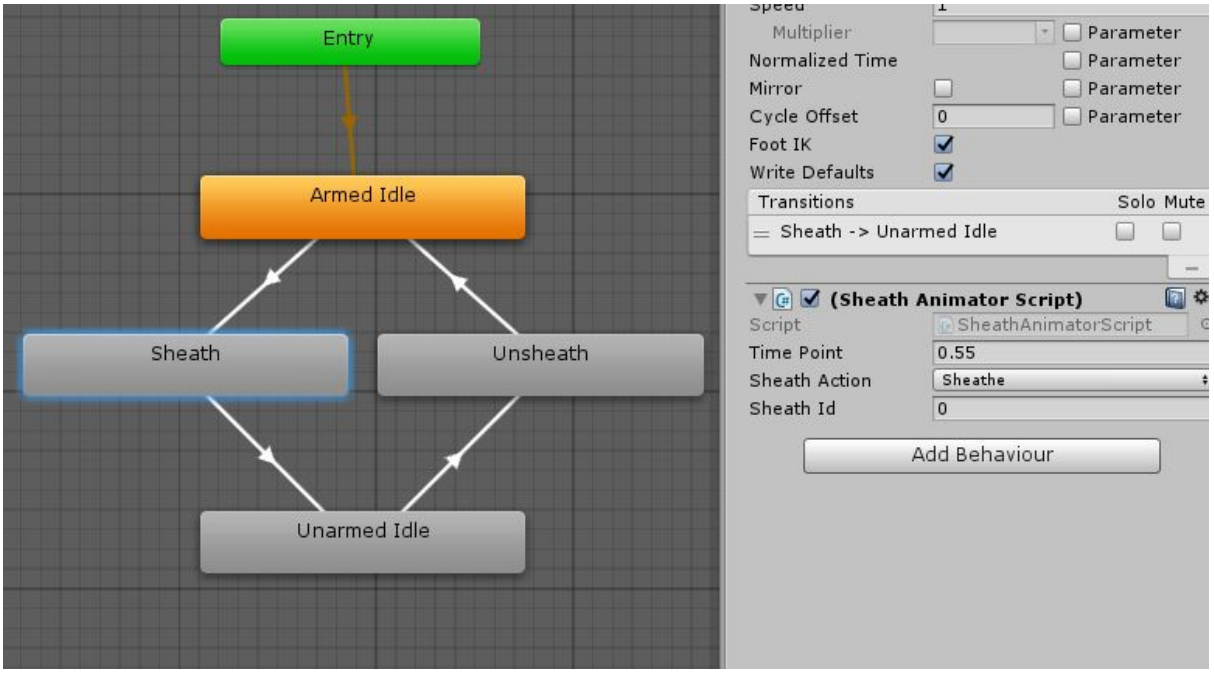
You will need to add this script to each animation state with the sheathe or unsheathe animations.

Sheath Animator Script fields

- **Time Point.** This is the point of the animation that will the sheathe or unsheathe occur. This is percentage (%) value, 0.5 means in the middle of the animation. Adjust this value if the character sheathes or unsheathes too early or too late. A tip for getting the exact % point is to use the animation preview window which shows the current time, frame and %.

- **Sheath Action.** If this is a sheathe or unsheathe animation.

- **Sheath ID.** The number of the sheath entry in the SheathComponentScript to be affected. If you only have one sheath entry this will be 0.



Example of a good configuration of the SheathComponentScript

Sheathe and Unsheath using Animation Events

If you prefer to use Animation Events you can create them and call ‘Sheath’ for sheathing and ‘Unsheath’ for unsheathing. The int values of the animation event is used as Sheath ID to determine which weapon will be affected (from the Sheath Entries of SheathComponentScript of your character).

Sheathe and Unsheath using your custom scripts

For sheathing or unsheathing from your custom scripts there are some functions you can use:

```
public enum SheathActions{Sheathe, Unsheathe};
public void Sheathe(SheathActions sheathAction, int sheathId);

public void Sheath(int sheathId);
public void Unsheath(int sheathId);

public List<Sheaths> sheaths;
public class Sheaths{

        public string sheathName;

        public Transform weapon;

        public Transform hand;

        public Transform sheath;
    }
```

Remember to add this line before your class:

```
using KevinIglesias;
```

CONTACT SUPPORT

For support, questions or suggestions regarding this product send me an email to:

support@keviniglesias.com

Kevin Iglesias - www.keviniglesias.com

- **More Unity Assets**
- **Facebook**
- **Twitter**
- **YouTube Channel**