

FIMPOSSIBLE CREATIONS

RAGDOLL ANIMATOR USER MANUAL

About Ragdoll Animator

- Ragdoll Animator helps **generate ragdoll rig** using unity built in **rigidbody** and **configurable joint** components. With the generated physical rig, ragdoll animator algorithm is applying forces to physical joints in order to **match ragdoll pose with currently played character animation**.
- Plugin offers **many ways to configure** ragdoll setup in order to work with every type of rig. You can make it work on humanoids, animals or use it to apply physics to single limbs like tail or single arm.
- Ragdoll Animator component is providing a **highly customized inspector window** (GUI) to help use it without confusion since there are a **lot of parameters to play with**.
- Package is providing many example scenes presenting different features which can be unpacked to project with "Demo - Ragdoll Animator" unitypackage file
- Ragdoll Animator offers many methods for custom usage through code. These methods are starting with "User_" like "User_SetLimbImpact". You can find a few more methods like this under "ragdollAnimator.Parameters.User_" as well as access for parameters you see in the inspector window, like ragdollAnimator.Parameters.
- You can combine Ragdoll Animator with my other packages, like Tail Animator. Check the manual pages for more details about it.

Contact and other links you will find in Readme.txt file

Index

1: Getting Started

- Demo Scenes and Overview (3)
 - Ragdoll Dummy (4)
-

2: Humanoid Ragdoll Setup

- Ragdoll Bones Setup (humanoid) (5)
 - Extra humanoid fields (6)
 - Generating Ragdoll (6-7)
-

3: Custom Ragdoll Limbs Setup (8-9)

4: Working with Ragdoll Animator

- Setup Category (10-12)
 - Play Category (12-13)
 - Extra Category (14)
 - Collision Events (15)
 - Physics Settings (16)
 - Coding Utilities (17)
 - Binding Arm / Leg to other joint with physics (18)
 - Solving Issues (18)
-

5: Using Ragdoll Animator simultaneously with other plugins (19-20)

1: Getting Started

Demo Scenes:

You can find demo scenes with many useful examples under:

“Fimpossible Creations/**Plugins - Animating**/Ragdoll Animator/Demo - Ragdoll Animiator/”
after unpacking the Demo - Ragdoll Animator.unitypackage file.

Adding the Component:

The component you will use is called **Ragdoll Animator**.

Select your character object on the scene, hit “Add Component” and go to “Fimpossible Creations -> Ragdoll Animator” or write “Ragdoll Animator” in the search prompt and select it.

Gizmos Overview:

After adding components to your character, **ragdoll animator is trying to find humanoid bones** out of your character rig, to speed up the setup process.

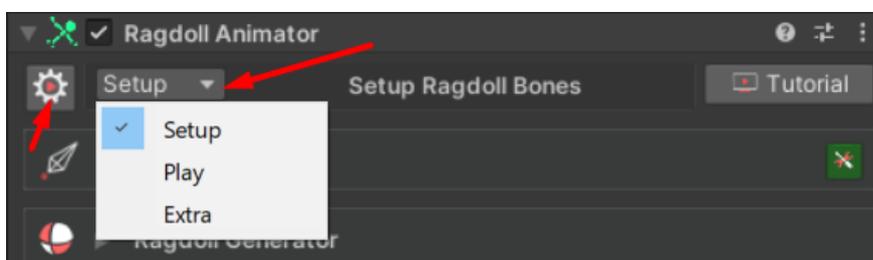
When you select an object with Ragdoll Animator on it, you will see **helper gizmos in the scene view** (“Bones Setup” needs to be unfolded) which can be very useful during setup and for debugging.

Inspector Window Overview:

Inspector window of **Ragdoll Animator is divided into few categories:**

- **Setup** - main settings for the physical ragdoll rig
- **Play** - playmode settings which defines ragdoll behavior/animation
- **Extra** - additional parameters for different use-cases

You can switch between categories using top-left view of the inspector window:



Handling Ragdoll Dummy:

By default after entering playmode, ragdoll animator generates a simplified copy of your character skeleton. There will exist an original skeleton driven by an Unity Animator and a second one called "Ragdoll Dummy" driven by Ragdoll Animator and physical components.

In order to separate animation and physics, it's required to use two rigs on the scene.

If you don't like to see "Ragdoll Dummy" in the hierarchy view (just for hierarchy view clarity - no performance change), you can enable "Hide Dummy" in Ragdoll Animator's "Extra" category.

If you need different types of colliders, for example box colliders for spine bones, switch to Bones Setup "Custom Limbs" mode instead of "Humanoid Limbs".

Pre-Generate Ragdoll Dummy:

Some projects may need to heavily configure a ragdoll dummy with custom components per limb / require ragdoll rigidbodies or colliders references.

Default settings are not giving the possibility to do this.

(you can still get target ragdoll bones by names - check "Demo_Ragd_PostAttach" demo script for reference)

To unlock editor mode configuration of ragdoll dummy, you need "Pre-Generate Ragdoll Dummy" mode.

To use it, you need to have already generated and adjusted ragdoll components on your character and have assigned "Root Bone".

Now you can go to the "Setup" category -> "Additional Settings" and turn on "Pre-Generate Ragdoll Dummy". Now you will see a generated ragdoll dummy in the scene hierarchy view you can work on.

("Pre-Generate Ragdoll Dummy" will be invisible until you assign root bone and generate ragdoll components)

After entering playmode, the ragdoll animator will use it as its ragdoll dummy.

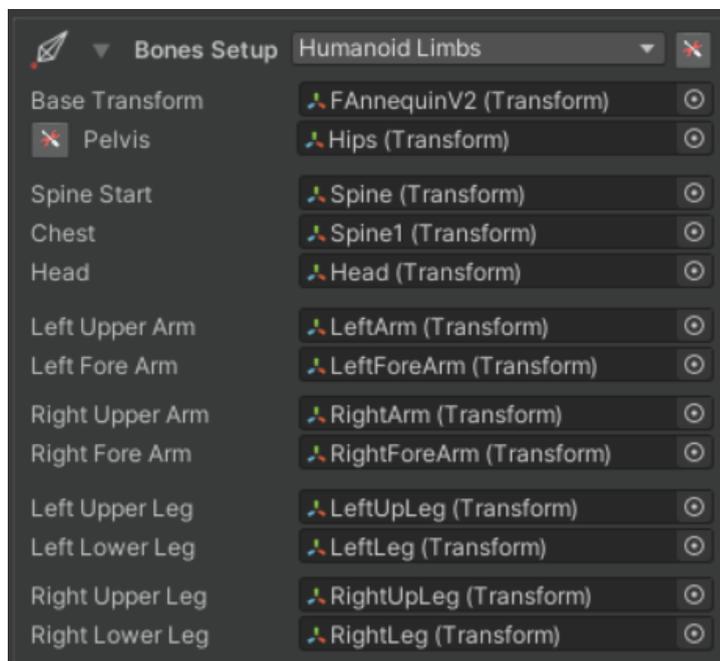
2: Humanoid Ragdoll Setup

Ragdoll Bones Setup (Humanoid):

After adding Ragdoll Animator to your character, ragdoll animator is trying to find the Animator component and get references to unity's humanoid bones, to prepare all bone references automatically for you.

If you're not using a unity humanoid rig, but generic or legacy rig, it will not work as precisely, but a ragdoll animator still will try to find correct bones searching them by names - **in this case it's required to verify if the algorithm found all the right bones**. In some cases your character model may contain leftovers from animation software which was included in the export. (Make sure ragdoll animator selected true bones - not the animator controls which are useless after model export)

That's how basic humanoid setup looks like:



All the fields you see here need to be assigned except "Chest" which is optional. You can set up a ragdoll animator on rigs with a single spine bone.

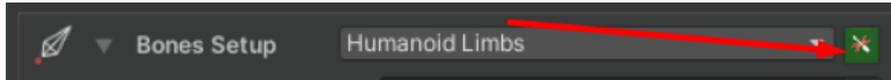
With "Bones Setup" unfolded, you will see more helper gizmos in the scene view, you can disable them simply by folding in "Bones Setup" view.

Extra Humanoid Fields:

If you want to match your character animation with more precision, it's important to include shoulder bones in the ragdoll dummy.

Against what it seems, shoulders are very important and including them in the ragdoll can make ragdoll animate much better.

In order to see extra humanoid bones fields, you need to hit tool button on the right to bones setup bookmark:

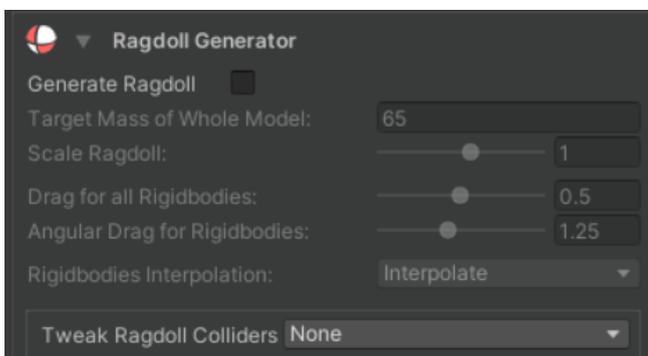


Then you will see new fields for shoulders, fists and feet.

Generating Ragdoll:

With assigned all needed bones, you can go to the "Ragdoll Generator" foldout.

By toggling "Generate Ragdoll" you will add rigidbody and physical joint components to the bones from the "Bones Setup" foldout.



With "Generate Ragdoll" toggle enabled, every change done in the inspector window will be applied to the character ragdoll colliders and rigidbodies.

With parameters below you can quickly adjust overall parameters for the physical setup.

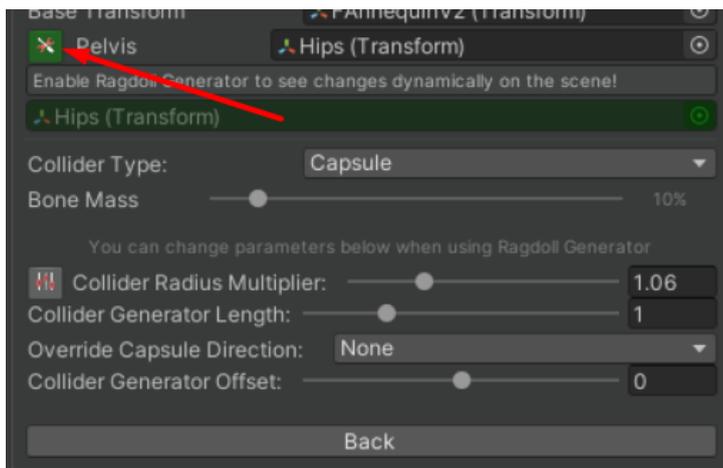
With "Target Mass of Whole Model" you can define the target weight of all rigidbodies together, and the algorithm will try to spread the mass value for rigidbodies according to human anatomy.

Next there is an all-colliders scale tweak slider and all rigidbody target parameters.

The default created colliders for ragdoll can be off and you may need to adjust them. Hit “Tweak ragdoll colliders” field and select if you want to enable movement or scale scene gizmos.

(if you don't see gizmos after enabling it, make sure your scene view has enabled “Gizmos” drawing, or reset editor layout - upper right corner of unity editor -> Layout -> Default)

If it looks like only the pelvis bone is off, you can do individual adjustment here using the tool button next to the pelvis field in the “Bones Setup” foldout.

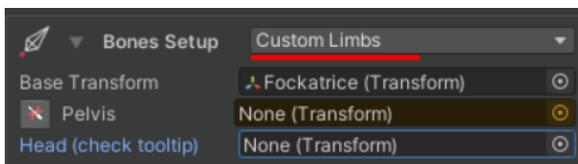


Here you can do adjustments which will be saved every time you generate a new ragdoll. You will see this view more often when using “Custom Ragdoll Limbs Setup”.

3: Custom Ragdoll Limbs Setup

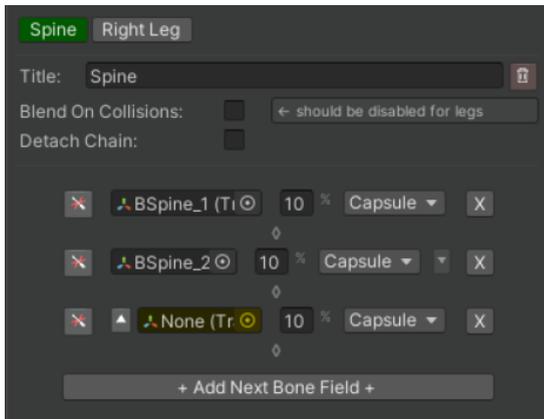
Custom Ragdoll Limbs Setup allows to create specific ragdoll setups. You can define how many limbs to use. If your character has 3 arms, tail, 4 legs then you can set up a ragdoll for it using a custom ragdoll limbs setup.

You can still use this mode for humanoids, if you switch back from humanoid setup to custom ragdoll limbs mode then the ragdoll animator will take care converting humanoid bone references onto custom limbs setup.



Custom limbs setup still requires two bone references before working with it. The pelvis and head. With these two fields assigned we can go on.

Now add the first Bone Chain and name it.



Blend On Collisions is enabling this feature for the limb (see description in 4th part of the manual)

Detach Chain is feature dedicated for very rare cases (see description in 5th part of the manual)

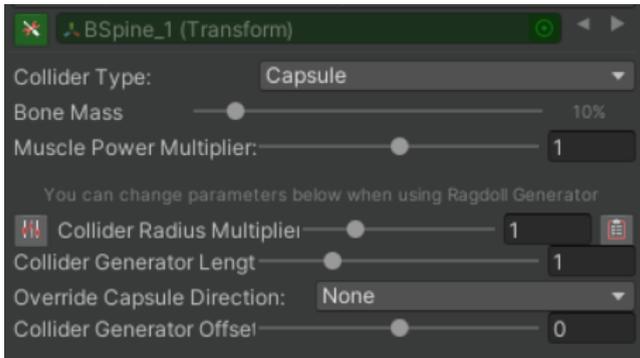
Assign the first bone of the target limb and then assign next bones in the chain one by

one by drag & dropping references using hierarchy view or use  button to quickly assign the first child of the previous bone.

The “%” field you see on the right is the percentage amount of mass which will be applied to the bone’s rigidbody. (enter it with mouse cursor for the tooltip)
If you assign mass = 100 in Generate Ragdoll’s “Target Mass of Whole Model” field then 10% of 100 will be applied to the generated rigidbody for the single bone.

Next is the type of the collider to use with the target bone and “X” button to remove bone from the chain if you need to re-arrange the chain.

Now the  button: it's switching you to the detailed setting for the selected bone.



Here you can adjust the same settings which were visible in the previous view and a few more.

Using ◀▶ buttons on the top right, will quickly switch between chain bones.

The **Muscle Power Multiplier** is giving unique muscle power for the single bone.

Muscle power is basically a multiplier for the Ragdoll Animator Spring value.

Next four fields are related to the size and shape of the ragdoll colliders.

It's recommended to turn on **Generate Ragdoll** under "Ragdoll Generator" to see changes in the scene view colliders gizmos.

Hitting  button removes sliders limitation and you can write any values under fields like "Collider Radius Multiplier" etc.

Hitting  button will apply collider adjustment settings to all bones in the chain. It can be useful when you're creating long chains like 10 bone tail chains.

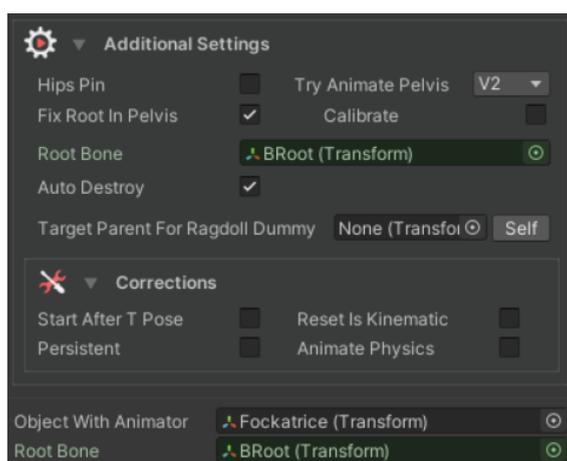
If you forgot these buttons functions, just enter them with a cursor and you will see a tooltip.

Hit the "Back" button on the bottom or  button to switch back to bone chain view.

4: Working with Ragdoll Animator

Setup Category:

After setting up your character bones and generating ragdoll, you see the rest of the view for the setup category.



First let's focus on the **Root Bone** field, since it's part of the bones setup process and it's an important field.

Root Bone is a reference to the first bone of your skeleton. It's important because the generated ragdoll dummy is using this bone as the main reference for the ragdoll animator algorithm. If you leave this field empty, the ragdoll animator will try to find it automatically, but in some cases it can choose the wrong reference and it will

make ragdoll work wrong. If the bone is assigned incorrectly, the ragdoll dummy may be generated with the wrong scale or keep the wrong position on scene.

Object With Animator is just a reference to the base object of your character. It's useful if you want to keep the Ragdoll Animator component on different game object but still apply its algorithms on your model.

Now **Additional Settings** foldout.

Hips Pin is a feature which enables position joint on the pelvis.

This feature is not fully completed since it's stealing a lot of animator pelvis motion in favor of physical behavior. Thanks to this feature, character's pelvis is not frozen but reacts with the rest of the body. Check "S_RagdollAnimator_Demo_Impact_HipsPin" demo scene for better insight.

With hips pin enabled, you will see a new foldout under "Play" category called **Hips Pin Adjustments**. There you can tweak a little position spring behavior but when the feature will be fully completed, these parameters will be removed.

Animate Pelvis (only with Hips Pin Disabled) is enabling hips animator follow.

Thanks to that, ragdoll dummy animations during non-free-fall mode won't look so stiff. The Animate Pelvis V2 is the same as V1 but V2 is applying more force to the bone muscles during non-free-fall ragdoll mode.

Fix Root In Pelvis is applying extra correction calculations to the dummy which is helping some of the exception case setup skeleton rigs. If you experience some troubles with your character ragdoll, try enabling it (and make sure you assign Root Bone!)

Calibrate is helping handle physics on not animated characters. It also can be useful if just some limbs of the character aren't animated. If you experience no muscle force on some bones of your character, turn Calibration on.

Auto Destroy is enabling destroying generated ragdoll dummy when the ragdoll animator parent object is destroyed. (if ragdoll dummy is not already parented)

Pre Generate Ragdoll Dummy (visible when assigned Root Bone and generated ragdoll components) field is described in 1st part of the manual.

With **Target Parent For Ragdoll Dummy** you can define target parent for physical skeleton to which ragdoll dummy will be moved on start.

Now **Corrections** foldout.

Start After T Pose is applying a few frames delay before initializing ragdoll animator on start. It can be helpful when using a ragdoll animator with third party plugins.

Reset Is Kinematic is ensuring rigidbody components "Is Kinematic" toggle to be in correct state after start. It can be helpful when using a ragdoll animator with third party plugins.

Persistent is applying Don'tDestroyOnLoad() unity method on the generated ragdoll dummy. It can be useful if you keep your character existing for the whole game lifetime. (for coders)

If your Unity Animator component is using **Animate Physics** update mode, you should also turn on **Animate Physics** here in the Ragdoll Animator. (see more under Physics Settings part of this manual section)

Play Category:

Here you do the most important tweaks for ragdoll behavior.



Free Fall Ragdoll is allowing pelvis bone to fall resulting in character falling on the ground.

If you disable it, the character pelvis will not allow the character to fall and you can use it as an active ragdoll for different body parts. So Free Fall **OFF** = walking character with physical body. Free Fall **ON** = falling character.

Ragdolled Blend can be used to smoothly switch between ragdolled pose and animator pose.

If you're using **Blend On Collision** then with

ragdoll Blend = 0% blend on collision will overwrite this value for selective limbs.
(more details down below)

Rotate To Pose Force is defining overall muscle force put into limbs.

(main multiplier for the Configurable Spring described in the next part of the manual)

If you want whole character to become limp, change this value to very small value

(not recommended to set it to zero, it's better to keep it at around 0.02 value if you want "numb" state)

But if you want the character to use full power towards animator pose, you can set it to 1, but beware since you want some loose, so physics are able to bend your character limbs.

Unity Solver Iterations is changing unity physics operations iterations for more precision. Higher value can make joints behavior more stable but with more performance costs.

Blend On Collision is a feature which can be useful if you want to keep original animator motion intact (no ragdoll animation deformations) and use ragdolled state only when collision happens. For example: your character arm starts touching something -> blend on collision is smoothly blending in ragdolled state for arm.

(warning: it will enable ragdoll for the limb even if the ragdolled blend is set to zero!)

Collision ends -> blend on collision is smoothly blending out ragdolled state for arm.

During playmode under Blend On Collision foldout you will see debug fields displaying objects with which limbs are colliding.

Configurable Spring is basically a spring parameter for the Unity Configurable Joint components which are used by the ragdoll dummy. This parameter defines the

so-called "Muscle Power". Higher value -> movement is more sudden, lower value -> movement becomes weaker.

Configurable Damp is again all Configurable Joints parameter value.

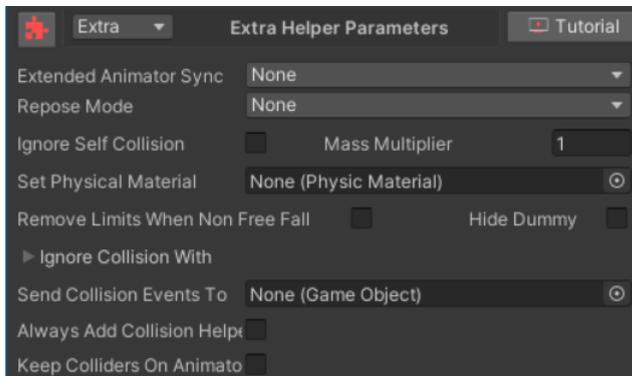
It's connected with configurable spring value and it's making the movement a bit softer. It's good to find values for these two parameters which works the best for your character setup.

Stand Up In Foot Point is a parameter which helps define target root position for the model in relation with ragdoll hips when triggering get-up animation procedure.

More Individual Limbs Settings, there you will find single limbs muscle power multipliers for some extra control.

Extra Category:

If you need more customization the “Extra” category can fulfill your needs.



Extended Animator Sync is trying to apply extra animation operations in order to make ragdoll or animator synced better with what you see on the screen. It was more useful in the previous versions of ragdoll animator, now it's not doing much and this feature may be removed in the future version of ragdoll animator.

Repose Mode is executed during Free Fall mode. It's changing the ragdoll dummy position to match the target character controller model position. It can solve many problems if you want your character to get up after being ragdolled.

Ignore Self Collision is applying Unity Physics Ignore selectively for each collider of the ragdoll dummy. If you experience some ragdoll bones stuttering, check if turning this parameter on solves it.

Mass Multiplier is multiplying the original mass of all rigidbodies of the ragdoll dummy on game start. It can be really helpful when finding perfect mass value for the rigidbodies for your character.

Set Physical Material is applying target physical material to all rigidbody limbs of your character. Physical materials are very important, consider using them.

Remove Limits When Non Free Fall will remove configurable joint rotation limits during animation matching mode. Sometimes the bone rotation limits can cause animation to look wrong, this toggle will give more freedom for physics animation matching.

Hide Dummy - Check 1st part of the manual.

Ignore Collision With with this list you can define if some scene colliders should be ignored by the ragdoll dummy bones.

With **Send Collision Events To** you can enable sending ragdoll limbs collision events send. (for coders)

Always Add Collision Helpers will always include extra components () on each ragdoll bone limb. Component contains information about ragdoll animator, which limb it is and other useful data for coders.

Keep Colliders On Animator will prevent removing ragdoll colliders from the animator skeleton when generating a physical ragdoll dummy.

Collision Events:

You can use ragdoll bones collision events to trigger custom actions when bones are colliding with something / when bones hits the ground during fall etc.

When you assign "Send Collision Events To" object, there will appear a brief description of what you can do to implement it.

One approach is using Unity's SendMessages.

On the "Send Collision Events To" object, you need to have an attached component with public methods like "**ERagColl**([RagdollProcessor.RagdollCollisionHelper coll](#))" for collision enter and "**ERagCollExit**([RagdollProcessor.RagdollCollisionHelper coll](#))".

The argument "coll" is a reference to bone info.

Ragdoll Collision Helper inherits from RagdollIndicatorHelper. RagdollIndicatorHelper is just bone indication info, RagdollCollisionHelper is implementing unity collision enter/exit methods.

The second approach is implementing the IRagdollAnimatorReceiver interface in some of the attached components. It will work faster than the "SendMessage" approach.

The RagdollIndicatorHelper contains information about parent ragdoll animator, it's ragdoll bone reference. (RagdollBone - in which you can find references to rigidbody, animator bone transform etc.)

If it's a humanoid bone you can read .LimbID to get the type of the bone.

If it's a custom bone chain bone then you can read it with CustomBoneChainName and two other bone chain related variables.

Physics Settings:

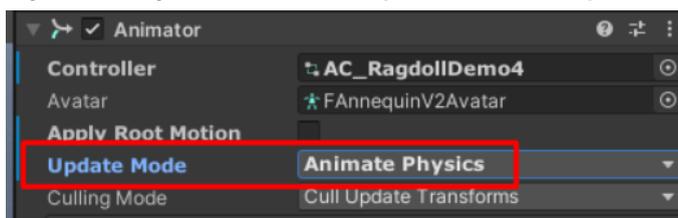
Unity Physics settings have a big influence on the Ragdoll Animator results. If your project can afford more performance for physics calculations, then it's recommended to change Fixed Timestep (Edit->Project Settings->Time) to value 0.01 it will give more precision to the physics.

Rigidbody mass and colliders shape defines how the physics are executed and limited. It's important to tweak these values if you want to customize ragdoll behavior.

Ragdoll Animator spring and damp defines how quickly a ragdoll dummy moves its limbs towards animator pose. You need to find "sweet spot" values for your ragdoll dummy.

The colliders' physical materials are also very important, especially if you want your character's arms / legs to be dragged during animation. If you assign lower friction physical components to the ragdoll dummy, the legs will not stop so hard on the floor during ragdoll animation.

If your Unity Animator component is using "Animate Physics" update mode:



you need to toggle on "Animate Physics" under ragdoll animator "Setup" category to be found inside "Additional Settings" -> "Corrections" foldout.

Coding Utilities:

If you want to control ragdoll animator parameters through code, apply body impacts, fade in/out ragdoll, change muscles power, there are plenty of methods for you to use.

First you need a reference to the RagdollAnimator component.

It's under "FIMSpace.FProceduralAnimation" namespace, so when you define variable you can write `FIMSpace.FProceduralAnimation.RagdollAnimator`, or write using `FIMSpace.FProceduralAnimation`; on top of your script file to write just RagdollAnimator when defining variable.

Now with `RagdollAnimator` `ragdollAnim`; variable you get access to many `ragdollAnim.User_` methods to manipulate the ragdoll.

If you need more deep access you need to use `ragdollAnim.Parameters` variable.

The `.Parameters` is a reference to the brain of a ragdoll animator.

It gives you access to internal variables like each limb muscle multiplier, bone references and others.

Binding Arm / Leg to other joint with physics:

You can make your character hold some objects and still be ragdolled + animated. (take look at demo scene "S_RagdollAnimator_Demo" there are examples of handing characters) In order to attach foot / hand to something, you need fist / feet ragdolled bones. If you attach a lower leg / lower arm it will make it look like a knee or elbow is attached instead of holding the targeted point. Generating fist/feet ragdolled bones are described in the manual part above this (Extra Humanoid Fields)

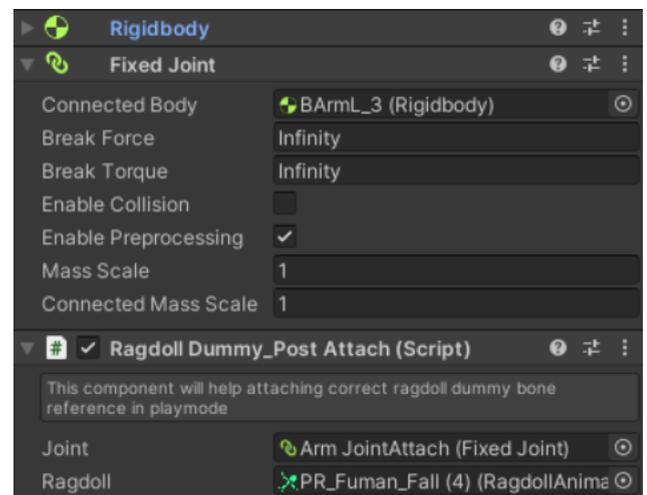
Now when you have required ragdoll bones, you need a target object to which you will pin the limb of the character.

Target object needs a **Rigidbody** and **Fixed Joint** component.

Now you need to assign our character hand reference to the Fixed Joint "**Connected Body**" field.

If you're not using a Pre-Generated Ragdoll Dummy, then you need a script called "Ragdoll Dummy Post Attach".

Assign here our Ragdoll Animator reference. Now entering playmode hand should be pinned with this target object.



Solving Issues:

For some tips about solving physics issues you can check the readme.txt file. There is a description for different cases you may be experiencing.

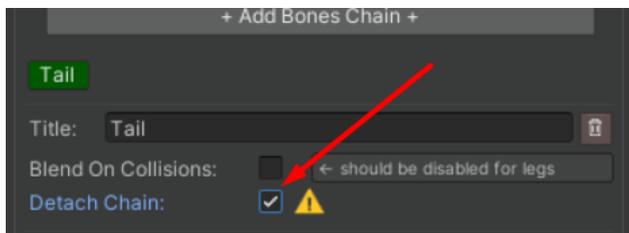
6: Ragdoll Animator simultaneously with other plugins

- **Using Ragdoll Animator with Fimpossible Procedural Animation Plugins:**

If you want to add world collision physics for your creature tail, you can try using Tail Animator and Ragdoll Animator at the same time.

Ragdoll Animator tail needs to be set up with “Custom Limbs” mode.

Tail limb needs toggle “Detach Children” in order to give physical joints more freedom in executing motion, the transform parenting is making physical tail animation look wrong.



The ragdoll animator tail limb motion will not look the same as Tail Animator original motion, since there is physics involved now and the motion is highly dependent on the rigidbody mass/drag settings, tail colliders size/physical materials, unity physics settings and more on. You will need to do more tweaking for best results.

- **Changing Ragdoll Animator's Execution Order:**

It's a really rare case, but some plugins can be working with execution order which overrides a ragdoll animator's job.

You can think about it like layers - first: Ragdoll Animator Animation Layer, next on top: other plugin animation layer which replaces all jobs done by a ragdoll animator. To prevent it, you can change the ragdoll animator execution order to be executed as last one. To do it, simply open RagdollAnimator.cs file and on the top you will see line: [\[DefaultExecutionOrder\(-1\)\]](#)

Change value -1 to for example 10000 to make sure it's executed as last.

- **Using with Third Party Movement/AI plugins:**

There are not yet dedicated implementations of Ragdoll Animator and popular asset store AI plugins, but in most cases Ragdoll Animator should work with them. The downside is that you will need to control ragdoll animator through code to enable/disable it and do specific impacts. You can still use UnityEvents without coding but sometimes it will not be enough.

Some of the AI plugins will detect generated ragdoll and set all rigidbody to kinematic causing character launch to space. To prevent it, you can enable "Reset Is Kinematic" which is in the Ragdoll Animator "Setup" category -> Additional Settings -> Corrections.

If you like this package please visit my asset store page for more or write a review for this asset ;)