



TABLE OF CONTENTS

- Welcome to Full Sail
- Installing
- Quick Setup
- Custom Meshes
- Textures
 - Movement Mask
 - Reverse Movement (Mast Collision)
 - Furl
 - Furled
 - Thickness
- Texture Combine Window
- Helper Scripts
 - Sail
 - Sail Group
 - Sail Furl
 - Sail Bounds
- Shader Params
- Wind Control
- Impacts
 - How it works
 - Ripple Texture
- Attach Objects to Sails
- Some Tips
- Version Changes
- Tutorial Videos

WELCOME TO FULL SAIL

Thank you for buying our Full Sail Asset for Unity. Full Sail is a shader based system that will add great looking wind effects to your ship sails, castle banners or other cloth type objects. All the animation and deformation is done via a shader making it super fast in any project. Easily have 100's of fully animated sails in your scene.

Full Sail offers a wide variety of options to control the look of your sails. Sails can taper in or out, be sheared. The sides can be arched or even be told to arch only when the sail is full of wind. You have complete control over how the sail is fixed to any object and also how the sail moves when the wind blows. How much the sails ripple and the type of ripple is also controllable.

Another advanced feature especially for ships is the ability to have the sails furl themselves away. Again you

have complete control over how this happens via simple textures.

You can apply emblems to the sails, and you can control if that emblem appears on the back of the sail easily as well. Also you can add color the emblem easily and add emissive color to have the emblems glow if you wish.

If your project means sails can get damaged then we have that supported as well, each sail can have its own damage settings and textures, so holes can appear, or sails can look shredded.

The shader also supports bump mapping and translucency so your sails can look great with the sun behind them.

Full Sail comes with some sail meshes included to get you started or you can add your own. Also included is a set of sail textures, damaged sail

textures and various controlling textures for attaching the sail and for the furling effects.

Full Sail can also be used for other simple cloth type objects such as banners, washing hanging on a lines and so on.

Full Sail has versions of the shaders for Builtin, HDRP and URP pipelines. Shader graphs for HDRP and URP are also

included along with all source code. Also included are some helper scripts to show how you can control the sails in your game project.

If you check the image below of the Sailing ship all the sails on the ship are the same mesh and material, all the different shapes, colors, styles, effects are controlled by shader overrides and material property blocks.



A ship fully rigged with sails

INSTALLATION

By default Full Sails includes the shaders for the Built In pipeline. You will find two versions of the shader for each pipeline. There is an optimized version and a normal one. The only difference is the Optimized version makes use of a single texture to control the wind movement, furling and furled sate, the non optimized version has a texture for each. This makes authoring a little easier as you can swap in and out different controlling textures quickly for testing and setup and when you are happy you can use the include combining window to create a

texture for use by the optimized version of the shader.

If you are using HDRP or URP pipelines you will find a couple of rar files in the 'Shaders/HDRP' and 'Shaders/URP' folders, they each contain a unitypackage with the shader graphs and shader source code for each pipeline, extract the unitypackage file to a folder and then go to the Assets menu and 'Import Package/Custom Package' and choose the file you extracted from the rar file.

QUICK SETUP

Included with Full Sail are some pre made sail objects. There are 4 of varying number of vertices, pick one and add it to a scene. You can at anytime change the mesh so you change and see how it

performs for you. Select the object and add the 'Sail' material to it. That is it, you are ready to play with the shader values to see what happens.

CUSTOM MESHES

Full sail comes with a set of Square sails in multiple mesh densities as well as a set of triangular sails which should be suitable for most situations considering the options the shader gives you for changing the sail shape.

If for some reason you can't get the effect you need from the included meshes you can add your own. There are a couple of rules for the mesh you should follow.

First it is important that the pivot point for the mesh be at the top centre of the sail object and that the forward direction is Z up is Y and across the sail is X. The shader expects meshes to be in the way.

Second the mesh should have a 2nd UV channel and that channel should be planar through the mesh so the UV values on the back of the mesh are the same as those on the front. The UV values are used to control the animation and furling of the mesh, so if the UV's are different on the rear the animations will not look

correct. The uvs should go from 0,0 in the top left corner to 1,1 in the bottom right.

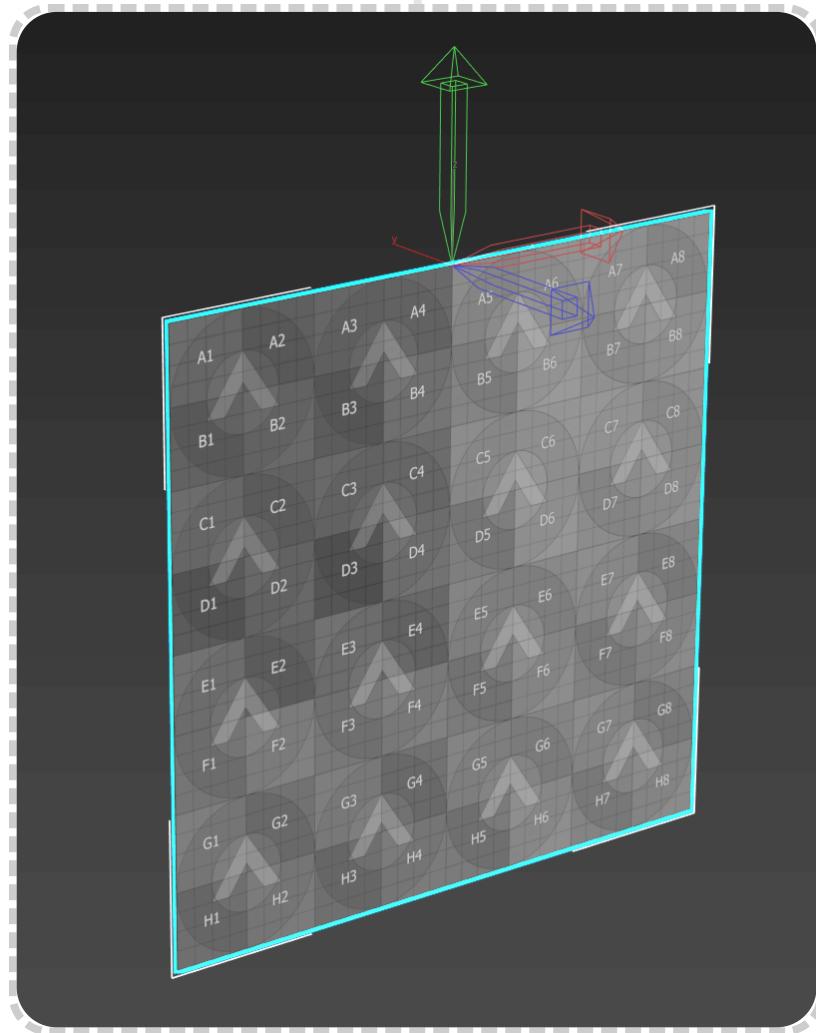
And finally the sail should not have any deformation, so it should be flat and the edges straight in general, you can have rough edges if you want but the shader can apply most of the shape changes needed so starting with a flat square mesh is the best starting point, once you get the hang of it you can play around with different shapes.

For triangle sails make the mesh a triangle shape to start with for best results.

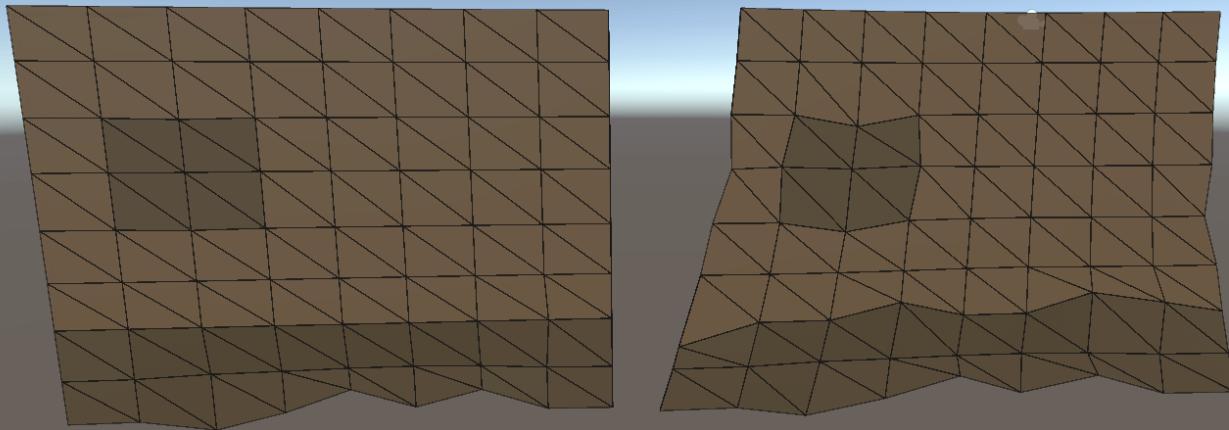
Another thing to note about the meshes is Unity will generate a bounds box for the mesh when it imports it. This is fine but if you have sails that deform a lot from their original position then it is likely that the Unity culling system will remove a partly visible mesh when it sees the mesh bounds box is not visible. To fix this the mesh bounds needs to be enlarged, there is a helper script called SailBounds included which

you can add to your sails which will expand the bounds enough so culling is correctly handled.

Once you have your mesh created just export as normal and add to your project and use the same as built in sail meshes.



A sail mesh showing UVs and pivot location. Note the image is showing the 2nd UV set, you can texture as normal using the first uv channel.

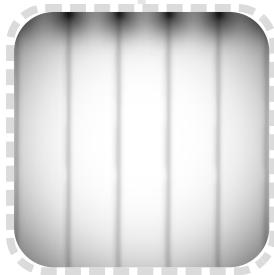


The Sail on the Right is a 3rd party sail, it has a shape which is useable but it is best to make the mesh with straight side edges as shown on the left and then use the shader values to change the end result shape. Note the sail has an undulating bottom edge, that will work fine.

MOVEMENT MASK

The movement of the sail is controlled by the Mask map. The lighter the area on this texture then the more the sail is allowed to move for that point. So if you want the top of the sail to not move at all then the top of the Mask map should be black. If you want all the corners to be fixed then again all the corners should be black. The gradients should be smooth for the best looking results. Check out the included examples to better get a feel for creating your own.

One thing to note when making a new texture is that when it is imported sRGB should be turned off in the import settings so the texture is linear, and the texture wrap mode should be set to Clamp otherwise you will get bleeding at the edges causing artifacts in whatever system the texture is controlling. This goes for the Furl and Furled textures as well.



Movement mask with multiple locked points and lines down the sail

Full Sail comes with some example Movement maps, or you can create your own using your fav image editing

software. We have though also included the ability to make a movement map in Unity. If you go to the menu and click

Assets/Create/Full Sail/Move Map a new Move Map asset will be created in the project. This contains values to create a Move Map from scratch.

The inspector will show the texture as you change the values. You can set the size of the texture with the width and height values. The fix falloff value controls how far the corner fixing values go into the rest of the map. The Falloff power control how quickly the value falls off. The four Fix values allow you to set per corner how attached the sail is, a value of 0 will mean no movement, 1 will mean it will move the amount allowed by the rest of the move map values.

If there is a texture in the Base Map then that will be used as the starting point for the new texture, if you are creating a Reverse Move map to stop the sail going through an object then you should select the move map used by the sail here so the sail mount points and general movement will be the

same. Then you can add fills or blobs to limit the movement as required.

The Fill value can be used to vary the movement map across the map, so you can add ribs for example to the sail, again in the gradient white is full movement black is no movement. The repeat value allows you to add multiple copies of the gradient across the sail. There is also a vertical fill option that does the same as fill but the gradient runs the other way.

Lastly the two curve values control the main movement across the sail and down the sail, when you change the curve values you will see the texture update.

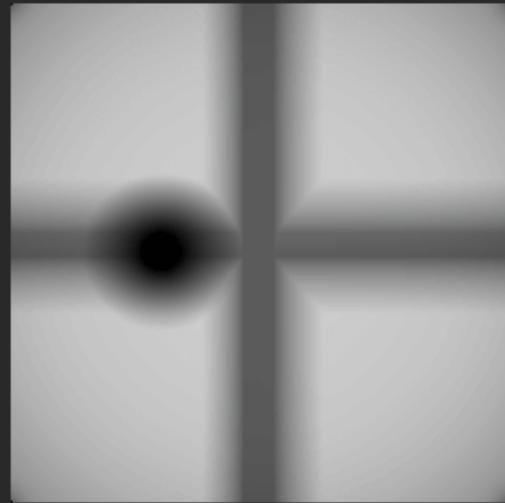
When you are happy with the values click the 'Save Map' and a texture will be generated in the Full Sails/textures folder and you can use this with the shader. If the texture is already present it will be updated so you can see your changes directly in the shader.

There is also a slightly more advanced option if you enable the 'Use Edge Curve' option then there will be four curves that control the movemap, one

for each edge of the map so you have more control over the map and can make some more interesting movements.



Move Map



Move Map Params

Width

Height

Base Map

Fix Falloff

Fix Falloff Power

Fix Top Left

Fix Top Right

Fix Bot Left

Fix Bot Right

Use Horizontal Fill

Fill

Repeat

Use Vertical Fill

Vertical Fill

Vertical Repeat

Use Curves

Use Edge Curves

Fill Curve X

Fill Curve Y

Use Blob

Blob Amount

Blob Pos X Y

Blob Radius

Blob Falloff

Blob Falloff Power

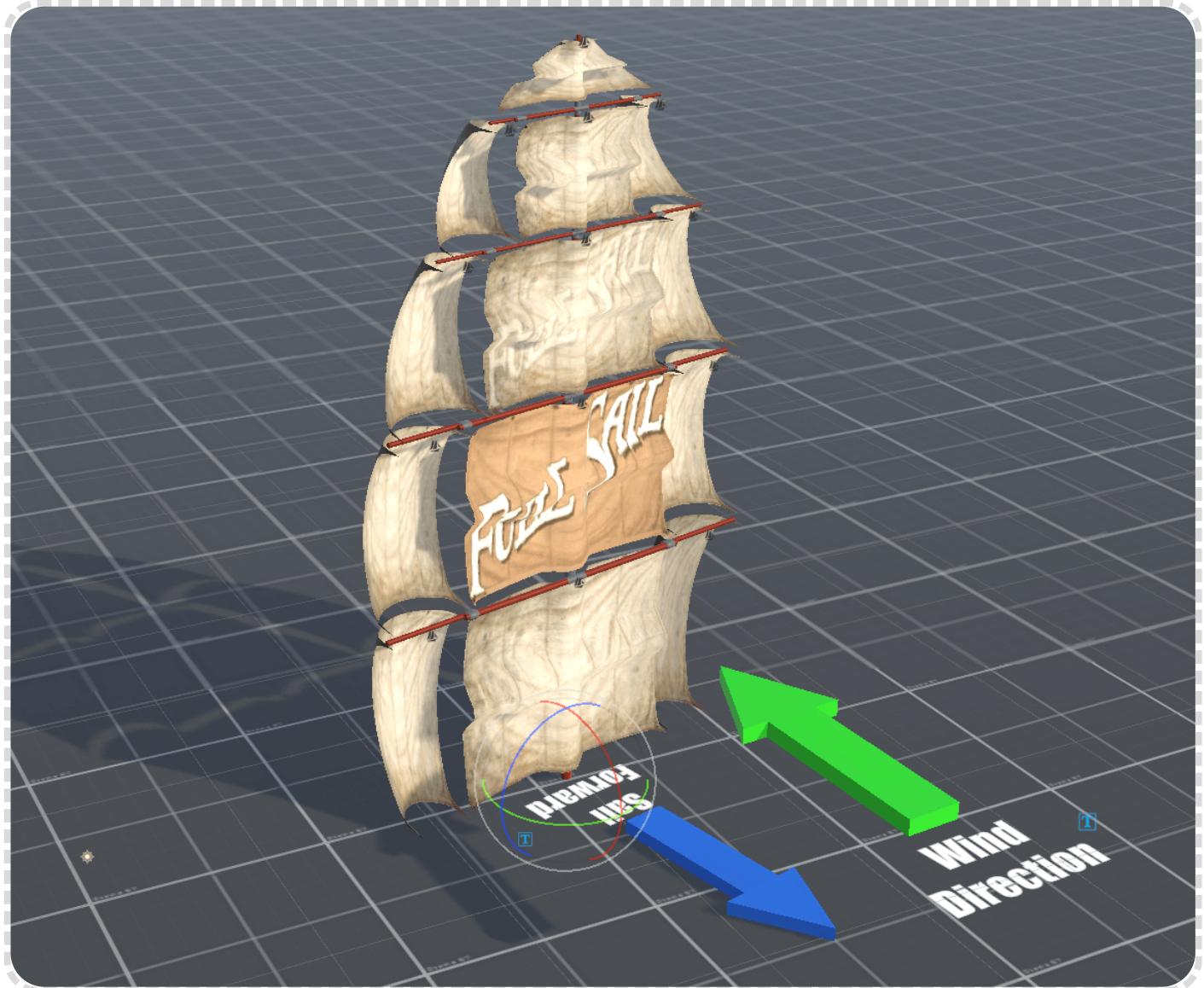
Save Map

REVERSE MOVEMENT

You can use the same Move Map for the forward and backwards movement of the sail, but for a lot of sails there can be an obstruction such as a mast behind the sail so it will look odd for that sail to move through the mast.

To stop this we added a Move Map to be used when the wind is against the sail, this

allows you to say duplicate the map you are using for the sail and add an area of reduced sail movement, this will allow unobstructed areas of the sail to move freely but blocked areas to stay the right side of the sail. When using a reverse move map it is a good idea to reduce the Sail Reverse value for the sail a little.



Middle sails use a Reverse Move Map with obstructions to stop the sail going through the mast

FURLING TEXTURE

How the sail furls is also controlled via a texture, with this texture darker areas will furl before lighter areas, so the top of the texture should be white so the top of the sail furls last. Again look at the

included examples to see what difference they make to the animation.

In the example furling texture below the bottom of the sail will furl start to furl first and the curved lighter lobes will mean those parts of the sail

will furl a bit later, so the expected animation from this will be bottom moving first,

then the edges and centre following a little behind resulting in a draped effect.



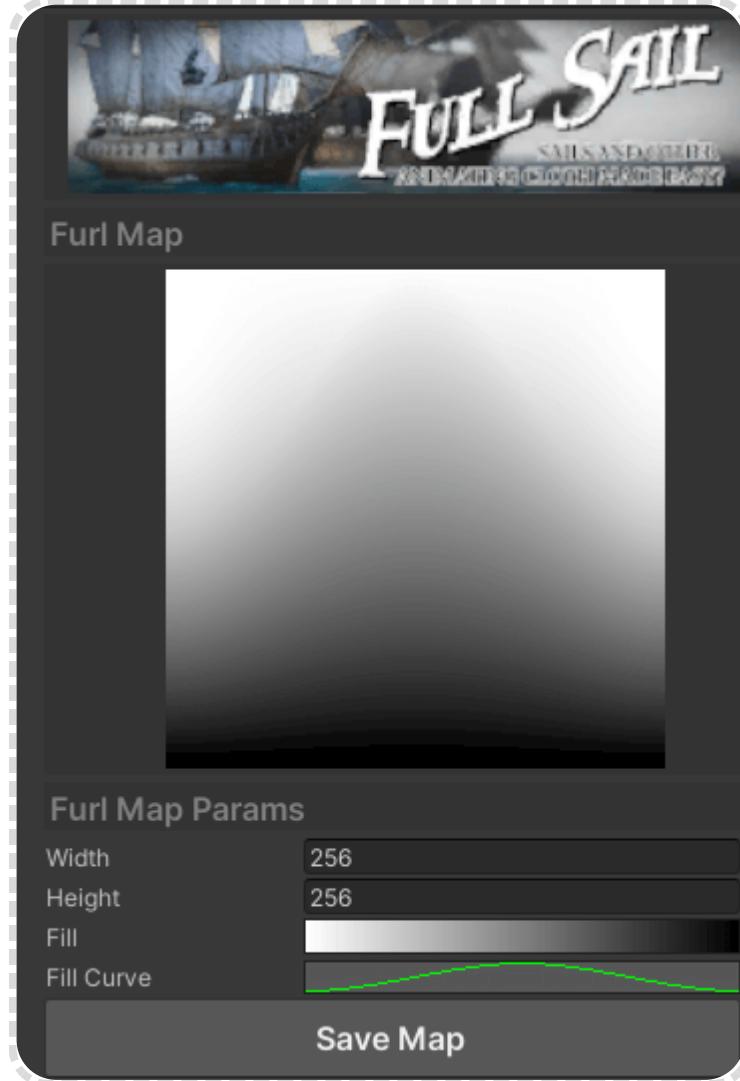
Example of a Furl Map

You can also create a Furl Map asset in your project from the Assets full sail menu. This creates a Furl Map asset in the project and allows you to build a new furl map by changing the values in the inspector.

You can set the width and height of the texture. The fill gradient is the base color, usually you should start with white at the top and black or

dark grey at the bottom. The curve values then allow you to add extra movement that controls how the sail will furl up.

When you are happy with the look click the Save Map button and a texture will be created in the Full Sail/Textures folder, you can then use that texture with the shader or in any overrides you want.



The Furl Map Inspector

FURLED TEXTURE

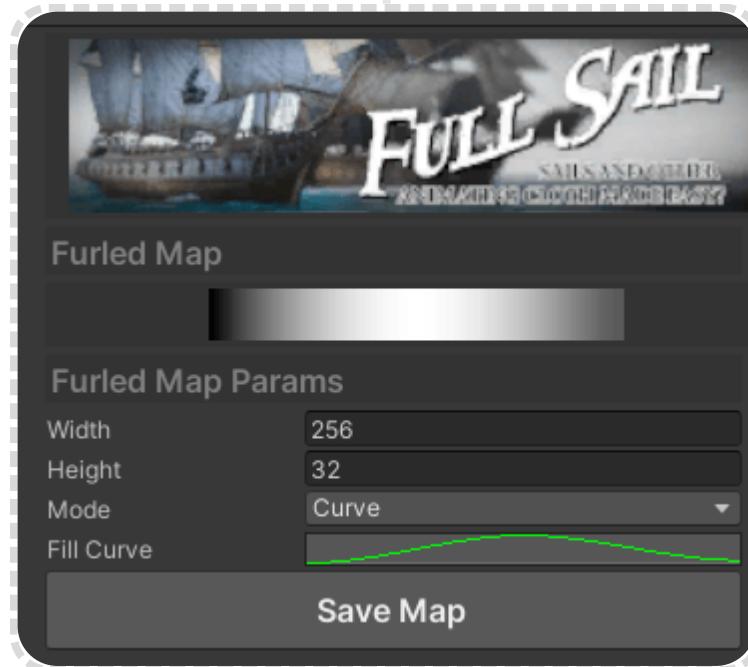
When the sail is fully furled its shape is controlled by this texture and the furled radius value in the shader. With a white texture the furled sail will form a perfect tube with the radius of the Furled Radius value. If you add darker areas then the radius

will be reduced accordingly. This allows you to get natural bulging looking furled sails.

As with the other textures you can create a furled map texture by creating a Furled Map asset in the project. Once added you can use the inspector to change the texture. Again width and

height can be used to set the size of the map. There are two modes you can use to create the texture, first is Curve based so by editing the shape of the curve will update the texture. White will mean the

furled sail will be larger than darker areas in the texture. The second mode is Gradient based, the curve mode gives more freedom but the option is there.



The Inspector for the Furled Map creator



Ship with all its sails furled

THICKNESS TEXTURE

This texture controls the thickness of the sail for the translucency, lighter areas in

the texture will let more light through.

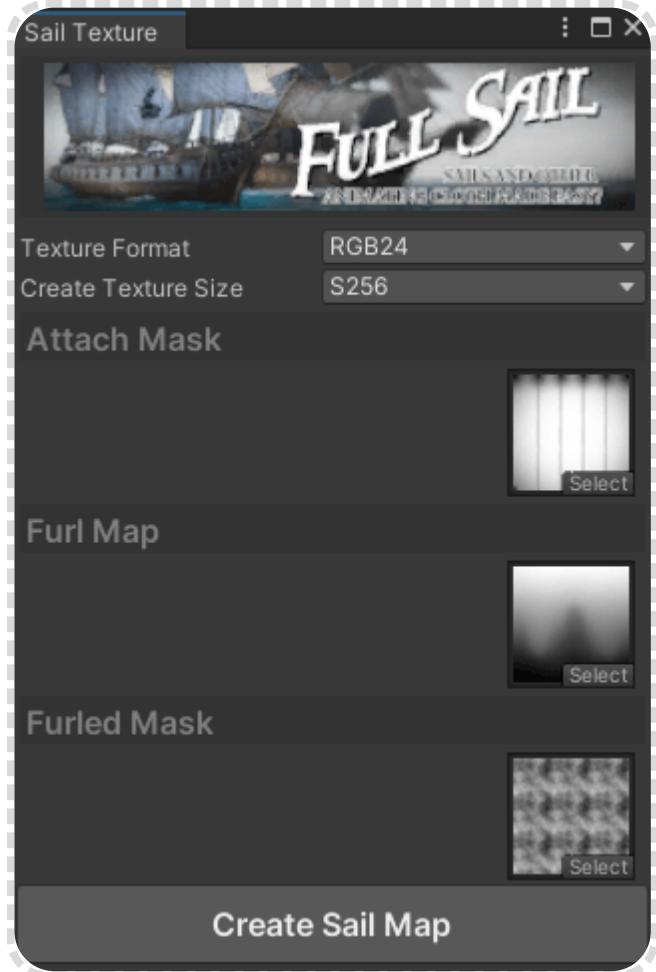
TEXTURE COMBINER

If you want to use the slightly optimized version of the shader then that makes use of combined Mask, Furl and Furled textures to reduce texture lookups in the Vertex shader. To make it easier there is a helper window in 'Window/Full Sail/Create Sail Texture'. When selected you can choose the Mask map, furl map and furled map. Click the Create texture button and new texture with the name of the mask texture with _Sail added to the end will be created. This has the Mask in the red channel, the furling in the green channel and the furled

in the blue channel. You can off course make these in Photoshop or other software if you want.

The Texture Format drop down allows you to select the format the new texture will be in, it is best to leave it on RGB24.

The Create texture size option allows you to pick the size of the new texture, the animation control textures do not need to be very big so use the smallest size until you notice any odd behaviour in the end animation.



Texture Combiner Window

HELPER SCRIPTS

Full Sail is a shader system so you can use it just like that and manage the values like you would for any other material in your game project, so for example if you have 10 sails in your scene you could have 10 different materials each using the Sail shader and apply one to each sail and then change the material values directly.

But we have included a few helper scripts to make life a little easier if you want to control groups of sails or change one param only on one sail on a ship with lots of other sails.

Below are descriptions of the few helper scripts that are included to get you started. I am sure this will expand as we get different use cases

suggested. Do let us know if you need something different and we will look at adding

support for it.

SAIL

This script allows you to set per sail any of the shader params. This is done via an Override system. Any Override added will replace the value that is set in the shader. This makes it easy to complex setups with each sail having different colors, textures, emblems etc. without having to have lots of different versions of the material. The overrides are applied via a MaterialPropertyBlock. To have the overrides applied a sail must be part of a Sail Group, see below for more info on groups. You are free though to use multiple materials and manage the values directly yourself.

To use add a Sail script to your sail object that has the shader on it, the included sail object prefabs already have the script added. Once added you will see an 'Add Override' button. Click this to add a new

Override to the sail. When added you will see a check box, this allows you to disable any override at anytime. Next you will see a drop down, this is where you select which shader value you want to override, depending on which value you use the next item will be either a slider, float value, color selector or a texture selected, this is the actual value you want to use for the sail. At the end the button can be used to remove the override from the sail.

In the image below you can see the sail has two overrides applied, one for the Sail Taper and one for the Sail Shear.

You can edit multiple sails at the same time so you can add an override and set its value to all sails you have selected which makes rigging up ships with lots of sails a bit quicker.



Sail script inspector with two overrides added

SAIL GROUP

The Sail Group component allows you to combine multiple sails into a set for controlling. The Sail Obs section allows you to add which sails you want to be part of the group. If the list is empty it will be filled with all the child sail objects found under the group object.

The Sail Group component will apply any overrides found on the sails to the shader, once a sail is part of a Sail Group you will see any changes you make to the overrides in the scene view.

The Sail Group also has its own set of overrides, this make it easy to apply a group override to the sails instead of having to add an override to each sail. Overrides on the sail will take priority of any Group override. Add the overrides in the same way you did for the Sails.

In the example below the Sail Group has 6 overrides added, one of which, Emblem, is disabled so will not be applied. All the other overrides will be applied to all the sails in the group. Click

the 'Add Group Override' button to add a new override to the list.

You can also see all the sails the group values will change, you can add as many as you need to each group.

You can also use the sail group to automatically add an Animation offset to each sail in the group. This will stop all the sails moving in the exact same way, you can do this by adding an override to each sail

but this option makes it simpler. Check the 'Add Ripple Offsets' box to enable it, the seed value is used by the random number generator so change that to get a different set of values. The Anim Range min max slider allows you to set the range of the anim offset to add, if the values are close together then the sails will be quite close in animation which is probably a good starting point. A large range will tend to make each sail behave uniquely.



Sail Group Inspector showing selected sails and overrides

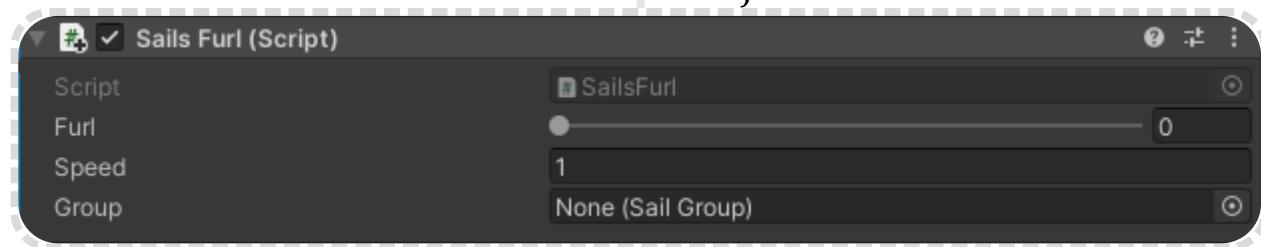
SAIL FURL

This is an example script to show the furling of the sails. When you apply it select a Sail Group that it will control. When you change the Furl value any Sail in the group that has a Furl override and a FurlOrder override will respect the Furl value here

and furl or unfurl as required. The furling is controlled by a SmoothDamp function so you can control the speed of the furling with the Speed value.

The furling order of the sails is controlled by the FurlOrder override for each sail. If your

group has say 4 sails, you can set all the furlorder values to 0 and then they will all furl together. If you set them to 1, 2, 3, 4 then they will furl one after another with the lower furlorder value furling before the others. You can easily setup furling groups with this or have sails that never furl etc.



Sails Furl Inspector

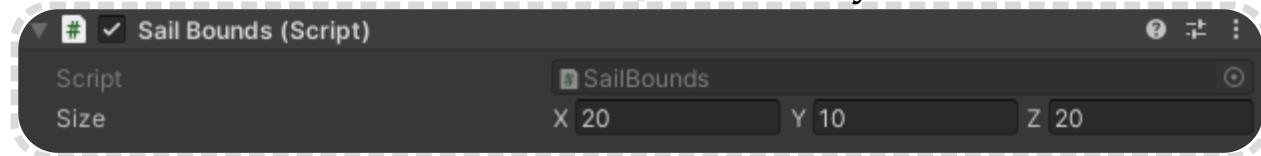
In the example below the speed of the furling is set to 1, lower this for faster furling increase it for slower furling. The furl value can be used to test the furling and directly controls the animation. The group option can be used to select a group to control, if left empty it will look for the Sail Group component on the object it is attached to.

SAIL BOUNDS

This script enlarges the bounds on the mesh to stop it being wrongly culled from the scene when it has a large deformation.

In the example below the script has been applied to one of the supplied sail objects

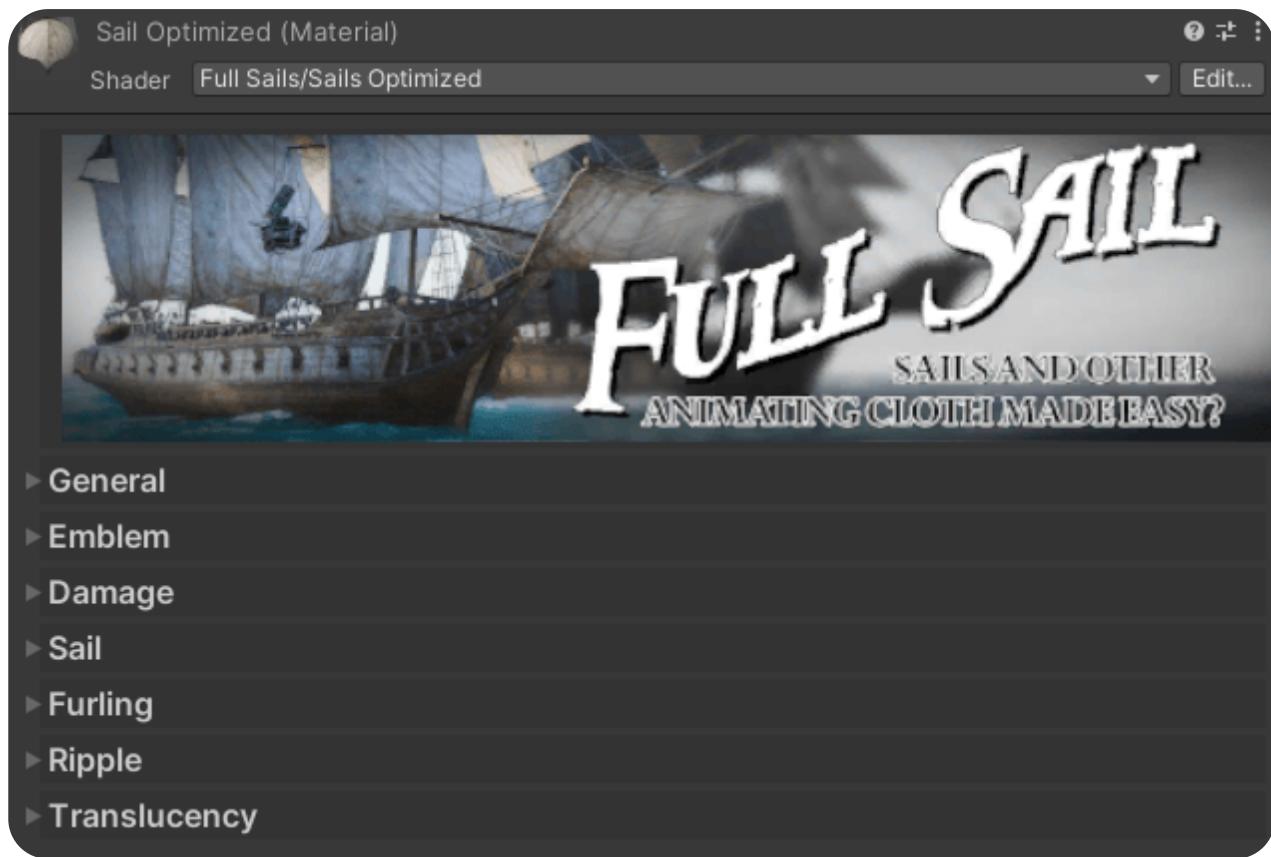
whose default bounds size is 10 10 0.02. The values increase this to 20 10 20 allowing for plenty of sail movement without Unity culling the object by mistake. Adjust these values when you have a sail at its fullest deformation to get the correct size of bounding box for your case.



Sail Bounds Inspector

SHADER PARAMS

The shader has a custom GUI to break up the various shader options into sections. The image below shows the various section, each section can be expanded by clicking on the Foldout bar. Below is a list of all the shader params and what they do. All the scripts and shaders have Tooltips that explain what each param does as well.



General Section



Color

Main Color tint for the sail. The same as any other material applies a tint the main texture for the sail.

Albedo

Main sail texture for the sail.

BumpMap

Bumpmap for the sail. Normal Unity bumpmap to add surface detail to the sail.

SpecColor

Specular color tint as normal, used with the Smoothness value below.

Smoothness

Specular smoothness value. Controls how much specular the sail has, best with low values as sails are not very shiny.

Emblem



Emblem

Texture to apply as an emblem to the sail. Emblems are applied to the centre of the sail. You can use the color value below to tint the texture or make it see through.

EmblemColor

Tint color for the emblem, alpha value for transparency.

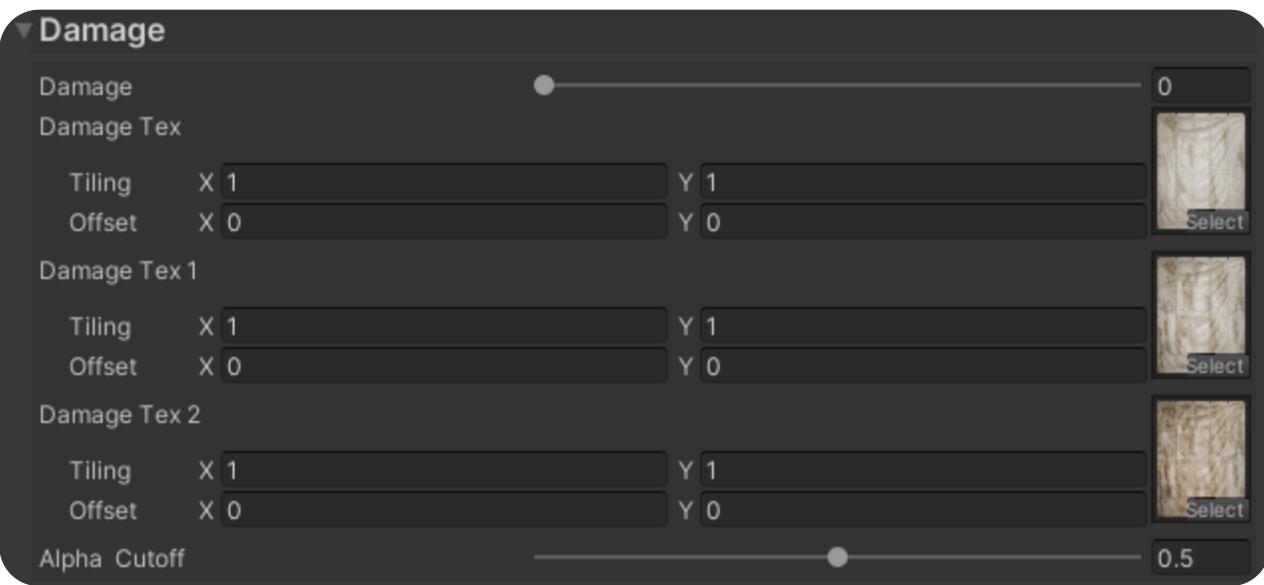
EmblemEMColor

Emissive Color for the emblem. You may want to have the emblem glow to make it look more magical, use this value to set that.

EmblemBackface

How much of the emblem is shown on the back of the sail. With the value at 0 then none of the emblem will appear on the back of the sail, if set to 1 then the emblem will fully appear on the back. If you use lower values then you can make it look like the emblem is just visible on the rear which can look nice.

Damage



Damage

Damage for the sail. 0 is no damage and 1 is full damage. The damage will work its way through the damage textures set below so you can have multiple damage stages. The damage is lerped in so will appear gradually as well, use the Alpha cutoff value to adjust how holes appear. The damage textures will need alpha channel to define holes or torn areas.

DamageTex

First stage of damage texture.

DamageTex1

Second stage of damage texture

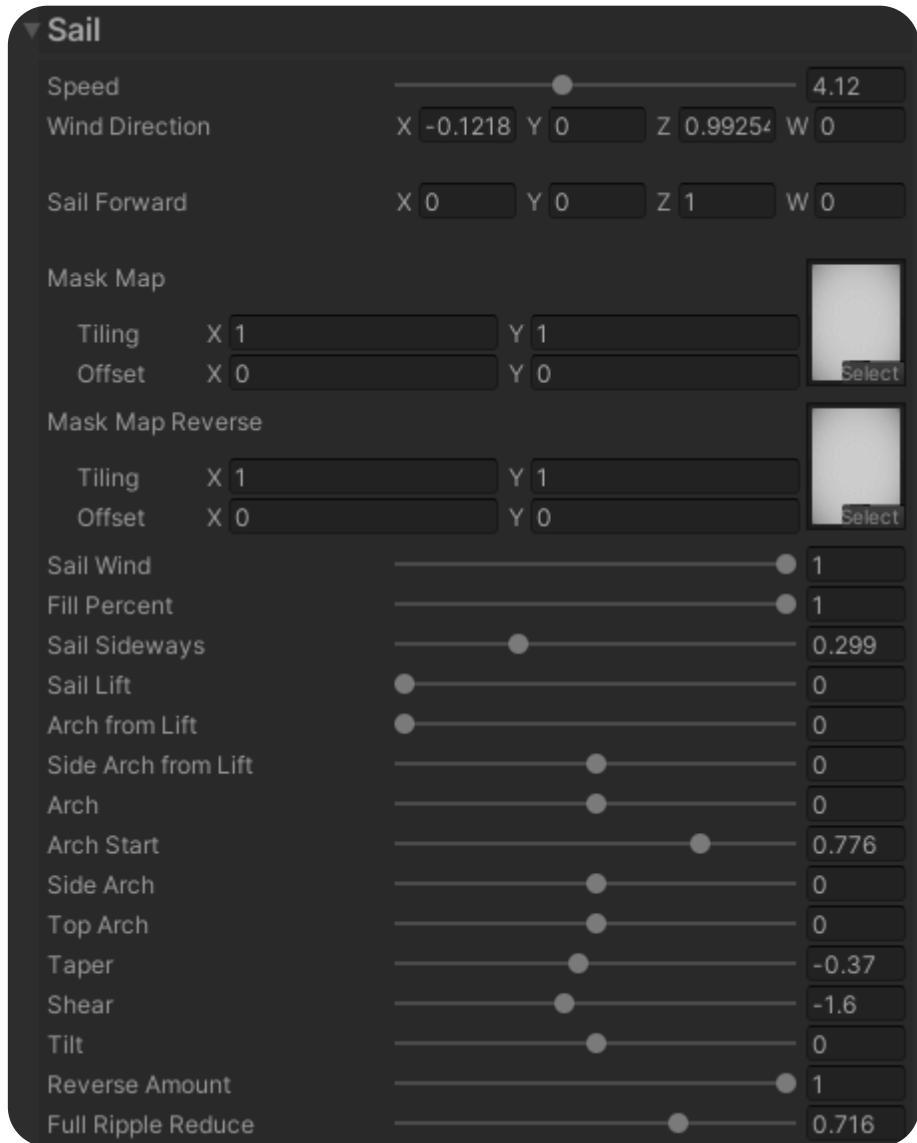
DamageTex2

Third stage of damage texture

AlphaCutoff

Alpha cutoff for the sail and damage. If the sail texture or damage textures have any alpha values this will control the cutoff for when they become transparent.

Sail



Speed

Speed of the wind for the sail. 0 is no wind and the sails will hang limp, as this value increases the sails will fill with wind and bulge and show ripples.

WindDir

Direction vector of the wind. Direction of the wind usually 0 0 1 0 to be blowing into the sails but you can change it. Will change the movement direction of the sail. The value is a direction vector the last w value is not used.

SailForward

Forward vector of the sail usually 0 0 1 0. Should usually be left alone, but if for some reason you want a different forward

direction you can change it here. Sometimes used is sails are leant at an angle to the wind. The value is a direction vector the last w value is not used.

MaskMap

Texture to control the sail movement (R), if using the Optimized shader then also Furl (G) and Furled (B). How this map is used is explained above, controls how far the sail will move with the wind.

MaskMapRev

Texture to control the sail movement (R) when the wind is against the sail, if there is nothing behind the sail then use the same map as MaskMap in here, but if you have an obstacle behind the sail then you can use a new map here to stop the sail going through the blocking object.

SailWind

The shader will compute the wind direction relative the SailForward value and this will be used to control how much the sail fills with wind, so if the wind is directly blowing into the sail from behind then the sail will move forward and billow out as expected, if the wind is 90 degrees to the sail then the sail will not move as the sail is not capturing the wind at all. If the sail is blowing against the sail then the sail will move backwards, the amount it moves backwards can be controlled using the SailReverse value below.

If this value is 1 then the full range of the sail movement will be used, if the sail is moving too much for your wind strength then reduce this value to reduce the amount of sail movement.

FillPercent

This value controls how much ripple is filling the sail, a value of 0 will add no ripple effect to the sail, a value of 1 will add all the

ripple. This can generally be left at 1 to get the full effect of the animation. Note the FullRipple value below can be used to automatically reduce the ripple effect as the sail gets stretched by the wind.

SailSideways

This values controls how much the sail will move with the wind as the wind becomes side one. A value of 1 means the sail is free to move fully, 0 will result in no sideways movement of the sail at all.

SailLift

Controls how much the bottom of the sail will lift as it fills with wind. If the sail is firmly attached at the bottom then set this to 0 then sail bottom edge will not move as the wind strength increases. If though the bottom is attached by say loose ropes then increase this, then as the wind gets stronger the bottom of the sail will lift. Use along with Arch values to get more realistic sail movement.

SailLiftArch

Controls how much the bottom of the sail will arch as it lifts. As the sail fills with wind and gets stronger you can use this to arch the bottom edge of the sail.

SailLiftSideArch

Controls how much the sides of the sails will arch as it lifts. Like the value above but does the same for the left and right edges of the sail.

SailReverse

If the wind is against the sail this will limit the movement, stops masks showing through etc. Normally you will have the wind

blowing into the sail from behind but if for some reason the wind blows the other way the shader will detect that and use this value to reduce the sail movement.

SailTaper

How much the sail tapers in or out at the bottom. Use this when setting up your sails to control the shape of the sail.

SailShear

Controls how much the bottom of the sail is moved relative to the bottom. Use this when setting up your sails to control the shape of the sail.

SailTilt

Controls how much the bottom of the sail tilts relative to the top, useful for some sail shapes. Use this when setting up your sails to control the shape of the sail.

SailArch

How arched is the sail, this is added to by any lift arch.

SailArchStart

How far down the sail the arch starts to appear. A value of 0 will start the arching from the top of the sail, 0.5 from halfway down.

SailSideArch

How arched the sides of the sail are, any side lift arch is added to this. You can have the sides curve in or out.

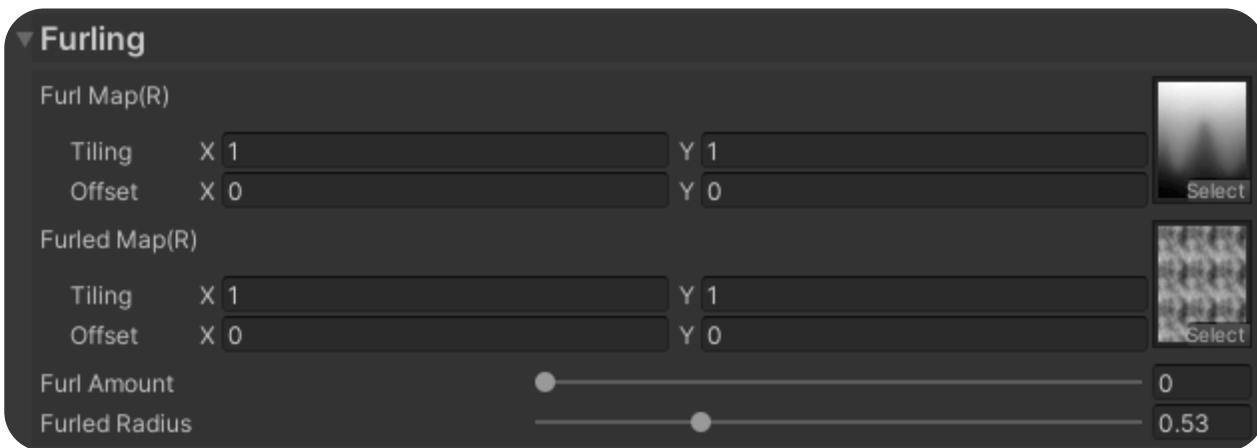
SailTopArch

How arched the top of the sail is. You can have the top curve up or down.

FullRipple

This value reduces the amount of ripple visible as the sail fills with wind. If this is zero then ripples will be applied to a sail no matter how strong the wind is, increase this if you want to see less ripples when the sail is being fully stretched.

Furling



Furl

How furled the sail is. 0 means the sail is full unfurled, 1 is sail is ready to be packed away and 2 the sail is fully furled.

FurlOrder

Controls the order the sail furls when in a sail group. Check the Sail Group section to see how the furl order value can be used to control the furling order of a group of sails.

FurlMap

Map that controls how the sail furls (R). See the Furl Texture section above for more info.

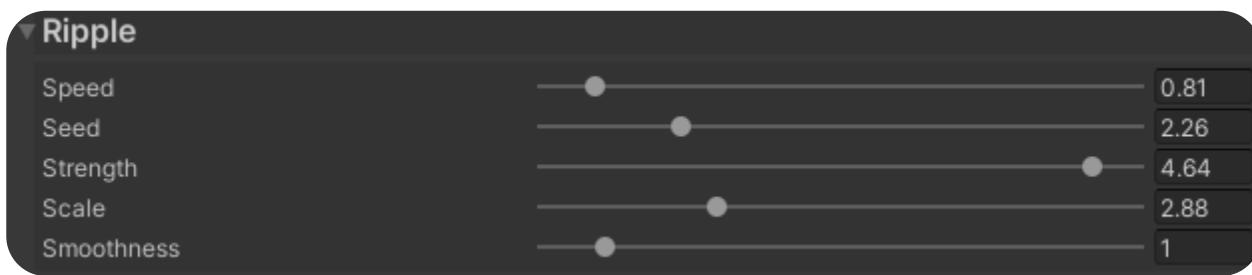
FurledMap

Map that controls the look of the fully furled sail. See the furled texture section above for more info.

FurledRadius

Radius of the furled sail. The radius of the furled state of the sail, this is multiplied by the furledmap amount.

Ripple



RippleStrength

How much the ripple effect moves the sail, gets used with wind value.

RippleSpeed

Speed of the rippling effect. How fast the ripples move across the sail.

RippleSeed

A value that is used to offset the ripple animation to stop all sails moving the same way.

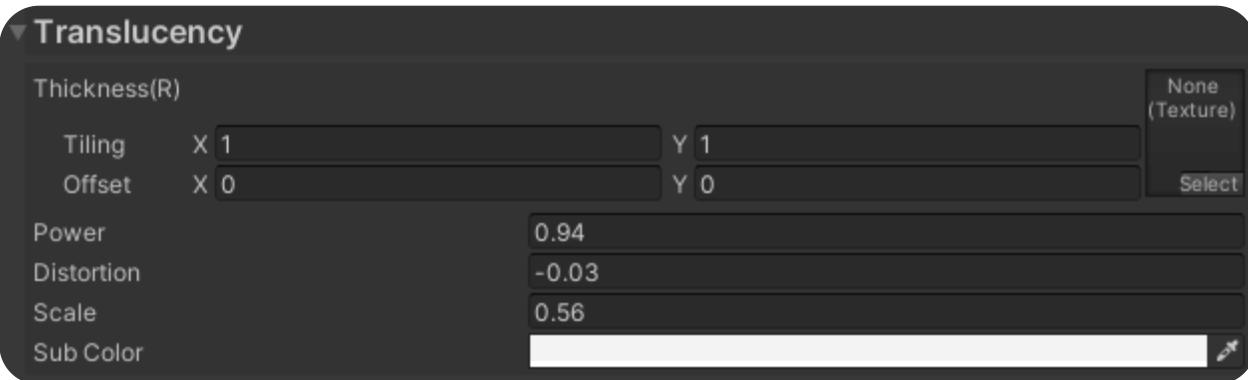
RippleScale

Size of the ripple waves. If this value is low then the ripples will be large subtle waves over the sail, as it gets higher the ripples will be come smaller and noisier.

RippleSmooth

Smoothness of the ripple waves. Smooths out the ripple peaks and valleys.

Translucency



Thickness

Texture used to control the thickness of the sail for translucency (R)

Power

Controls how the translucency falloffs from the light direction

Distortion

Translucency distortion

Scale

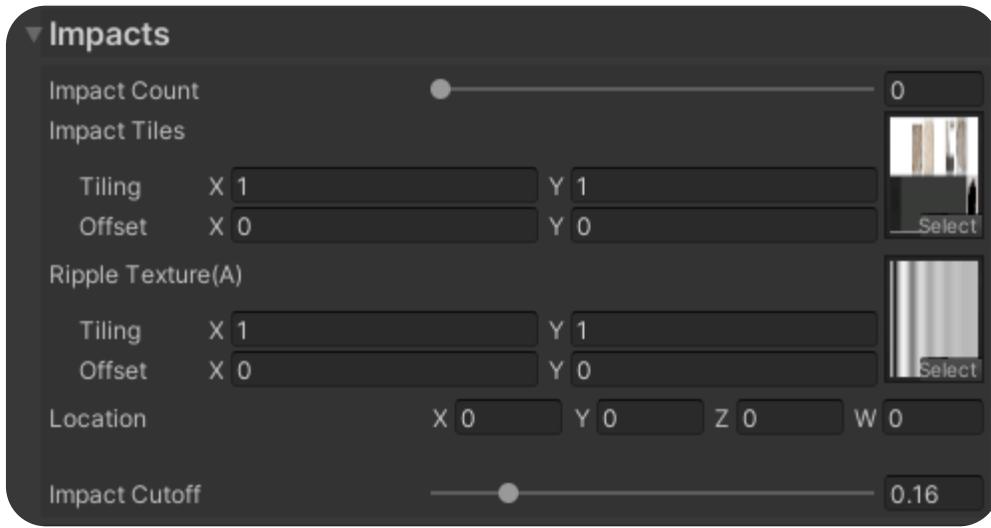
Adjusts the amount of translucency

SubColor

Sub surface scattering color

Impacts

Note: Impacts are not currently available on HDRP and URP as Unity has so far not added support for Vector4 arrays in ShaderGraph.



Impact Count

Number of impacts to show on the sail. This value should really be set from a script. See the Impacts section for more info.

Impact Tiles

This texture holds the 8 impact mark textures along with the 8 repair textures. You can select a type to use when you add an impact to the sail.

Ripple Texture

This texture controls the look of the Ripple effect, it is basically a wave form lookup and says how far to move a vertex over time. See the Impacts section for more info on this texture and the included helper for building your own ones.

Impact Location

This controls the ripple effect from an impact, the XY parts describe the UV location on the sail for the ripple origin. The Z value is the time when the impact happened, and the W value is the size of the effect.

Impact Cutoff

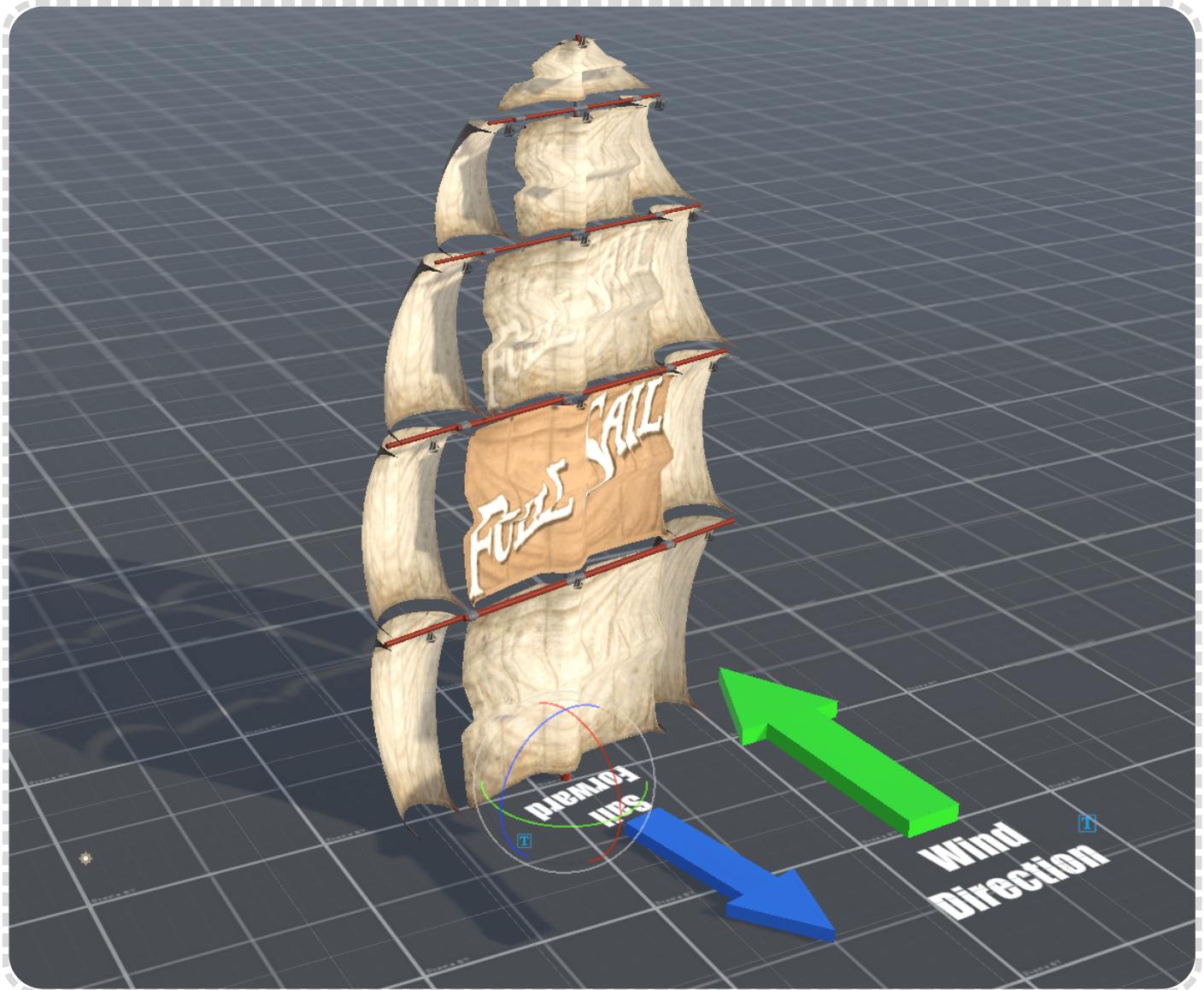
The impact marks are applied using a decal alpha cutoff system, use this value to adjust the cutoff point if you use your own textures.

WIND CONTROL

You will likely want your sails or cloth to react to a general wind direction. To do this set the 'WindDir' param to the direction of your world wind vector, the strength of the wind is set using the 'Speed' param. If your wind direction and speed is global then you can just set the values directly on any sail materials used and that value will be used across all sails. If you have different wind directions and speeds then it is best to set a Sail Group override for the

'WindDir' and 'Speed' values. Note for the WindDir value it is expecting a normalized Vector3 value so a wind direction along the Z axis would be a value of 0 0 1. The w component is not used in the WindDir param.

The shader will automatically compute how much of the wind direction is pointing into the sail and will adjust various params itself to control the sail movement. Check the param guide for 'SailWind' to learn a little more.



IMPACTS

How it works

The shader supports adding decal style impact marks to the sail. You can have up to 16 marks per sail (this can be increased) and each mark can pick from one of 8 types of damage to apply, and each mark can have a repaired

version so sails can be patched up. The marks can each have their own size as well.

To create a mark on the Sail there is a simple set of methods in the Sail Component.

AddImpact(float x, float y, float size, int type, bool remove = false)

The X and Y values controls the position on the sail to apply the mark, they work like uv coords so are normalized positions on the sail, 0,0 would be bottom left corner and 1,1 would be top right. The size value controls the size of the decal applied, a size of 1 would fill the whole sail, and the type is which texture mark to apply.

If you have filled the max number of impacts allowed per sail then adding a new impact will not appear. If you set the remove param to true then it will remove the oldest impact from the sail allowing the new one to be added.

PatchIndex(int index)

Calling this will turn a damage mark into its patched/repaired version. Index is the collision index. Indexes are the order the damage is applied to the sail.

UnPatchImpact(int index)

Will remove the patch from an impact and make it a hole again.

RemoveImpact(int index = 0)

This will remove an impact all together from the sail. If no value is passed the oldest impact will be removed.

Included in the assets are a couple of example scripts that show the process of adding impacts and ripples to the sails. ImpactBall.cs is an example of a script to apply to an object that you want to hit a sail. It detects when it is going to hit a sail and adds an impact to the sail at that point. Sails will need a mesh collider on them that is not convex as impacts make use of UV values to position the impacts. This may change and be improved in a later version.

The second script TestImpact.cs shows how impacts can be added directly to a sail from a script.

It is important to note that for ripples to appear on a sail that sail will need an ImpactLocation override applied to it.

AddRipple(float x, float y, float size)

Will add an impact ripple effect to the sail. X and Y describe the position and should be the same values that were used in the AddImpact. Size is the magnitude of the ripple effect. Only one ripple can be playing at a time on a sail so adding a new ripple will stop any currently playing ripple effect. And again for a ripple effect to play the sail must have an ImpactLocation override applied.

Colliders

The sail will need a mesh collider attached so that raycasts will work. One thing to be aware of is that the sail shader can deform the sail a fair way from the original mesh position and the mesh collider is not updated so if the sail is full of wind and stretched out the impacts will seem to happen behind the sail a bit and depending on the direction of the projectile the position may be off. If you use a custom proxy mesh that is inflated in the Z direction that will reduce those issues, the included sails have such an extended collider mesh applied.

The collider mesh only needs to be a simple box or simple shape to match the sail, but the uvs you be the same as the sail mesh as the uv value returned from the raycast dictates the position on the sail.

IMPACT RIPPLES

When an impact happens on a Sail you can also add a ripple effect to the sail to show the

impact. The ripple is controlled by a texture which says how far a vertex is to

move over time. It is basically a Curve turned into a texture lookup. We have included a 'Ripple Map' type into Full

Sail so you can create your own ripple effects to suit your needs.



The Ripple Map section in the inspector will show you the actual texture being generated from the params.

The Width value says how wide to make the end ripple map and the high is how high, currently you only need a texture 1 high for it to work.

Negate can be used to quickly reverse the directions generated from the Curve input. This saves you have to manually move every point in the main Wave Curve if you need to reverse the values.

The Amplitude value can be used to control the overall effect of the ripple, again this is an easier way of adjusting the output compared to altering each value in the main curve.

Scale allows you to adjust the X axis of the main curve to spread it out more or bunch it up.

The Ripple Wave Curve is the main value that controls the effect, in the image you can see a basic sin wave that decays away over time, this

should be good enough for most cases but you may want the ripple to die out quicker if so you would change the curve so it has less peaks and those peaks are smaller.

Save Map will write out the changes to the texture with the same name as the object you

are editing, this texture can then be used in the Ripple value of the impacts section of the shader. You can off course like any other shader param add an override to the sail so you can have different ripple effects for different sails.

ATTACH OBJECTS TO SAILS

There maybe times when you need to attach an object to a sail and have it move along with the sail as it ripples, fills with wind or gets an impact. You can do this using the Sail Pos component.

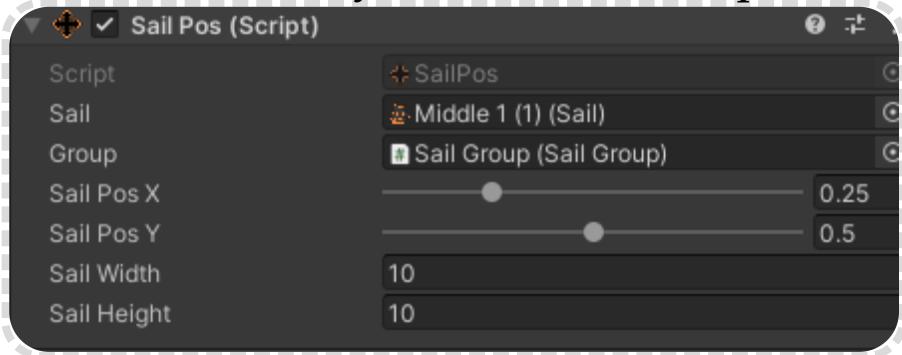
To use the Sail Pos system select the object you want to attach to a sail, this can be an empty gameobject or any other object and add the Sail Pos component in the usual way. Once added you will see a 'Sail' value, this is where you select the sail to which you want to attach the object. If that sail is in a Sail Group then select that Group in the Sail group value as well. This will mean the Sail Pos script can

access any sail params or sail group over ride values so it can calculate the position correctly. If the object is not appearing attached it is likely you have not selected the correct group.

If you are using the sails included with Full Sail you can ignore the Sail Width and Height values, but if you using custom sails then these values should be set to the width of the original sail mesh and the height. In the case of the default sails they are 10 units by 10 units. These values scale the normalized Sail X and Y values to get the correct local space position on the sail.

Once those values are chosen then the object should have moved to a corner of the sail, you can now either drag the object around the sail using the gizmo or you can use the

Sail X and Sail Y sliders to position the object. That's it the object will now follow the movement of the sail, this is great way to add rigging lines to your sail for example.



Sail Pos Inspector



SOME TIPS

When setting up your sails try to insure that the sail objects have no extra rotation applied so should generally be 0 0 0. You can off course rotate them around the Y axis for sails that run from front to back, and Z axis, but try to not to change the X angle.

If you are using the Sail component to change params on a sail then that sail must be in a Sail Group, if not the params will have no effect. If you don't want to use the Sail and Sail group system you can just duplicate the material and apply each new duplicate to it's own sail. You can then change the material values directly for that sail.

There are a lot of options in the shader so there is a bit of a learning curve as to what to use and when, so here are a few tips to help you get the most out of Full Sail.

When adding sails to your ship you can make use of the Scale gizmo to get the overall size of the sail right. Try not to scale in the Z direction though as this will result in exaggerated movement of the sail, the Scale Z value for your sail should be 1.

Also you can rotate the sail object to align the top edge of the sail to any beam or rope.

To get the shape as you want it use the Taper and Shear values to get the overall shape as you need, you can also use the Tilt value to adjust the angle of the bottom of the sail.

By default the sails are either Square or triangular with straight edges, you can add a curve to the sides by making use of the Arch values.

For more realistic sails try using the Lift value to raise the bottom of the sail as it fills with wind, and use the Arch

lift values to curve the sides of the sails as they fill with wind, this will give a much nicer look and feel to the sails and really make them feel like they have caught the wind.

If the sails have no mast behind them then make sure the Sail Reverse value is set to 1 to allow full movement when the wind is against the sail.

If there is an obstruction behind the sail then make use of the reverse Mask map option to stop the sail going through objects, this really makes the sails look part of the ship.

Full Sail comes with example textures for controlling the movement, attachment, furling, ripples etc. that work well but try making your own to get it exactly the way you need.

VERSION CHANGES

Version 1.16

- Fixed small issue in the Sail Pos system if no Reverse Mask Map was being used.
- URP Optimized Shader now has a Ripple Noise selector
- HDRP Optimized Shader now has a Ripple Noise selector
- Fixed some subtle differences in the various versions of the shader which led to Sail Pos being a little out of position in HDRP and URP

Version 1.15

- Small change to shader maths for Wind direction transforms.

Version 1.14

- Sail Pos system now correctly takes into account if a param is disabled in a group or on a sail

- Added missing _FurlMap to URP optimized shadergraph.
- Added missing _FurlMap to HDRP optimized shadergraph.
- Fixed the missing custom UI for the URP Shaders
- Fixed small bug in the sail pos where a param was labelled wrong which could lead to a wrong position
- Fixed an issue in URP and HDRP with the Sails not updating in the first frame due to OnBecameVisible executing in a different order to Builtin

Version 1.13

- Added an object attached to the sail in the demo scene to show how it is used.

Version 1.12

- Experimental Support added to HDRP shader for impacts, please report any issues
- Experimental Support added to URP shader for impacts, please report any issues
- Update URP and HDRP Shader Inspector so you can click the section headers and not just the arrow to expand the params

Version 1.11

- Shaders updated to keep a furled sail from being deformed due to sail scaling.

Version 1.10

- Changed the way ripples are applied in the shader, the ripples now happen in the sailforward direction which looks better than the old way.
- Moved some common shader functions to a cginc file for reduced shader sizes
- Sail Pos script updated to work correctly with the new ripples
- HDRP and URP shader graphs also updated to use new ripple
- Fixed the various create texture helper windows causing an error on Unity 2021+
- Added an Unlit shader for builtin pipeline which will run faster on mobile.

Version 1.09

- Standard and optimize shaders updated to use the new ripple noise texture system so will work correctly with the SailPos system
- Small change to the shader after a report of an error on some versions of Unity about missing _VorSeed param.

Version 1.08

- Sails will no longer update if they are not visible.
- Sail shader param setting now uses ID's instead of strings for faster updating
- Made the various map builders make the generated textures readable so you dont have to change them in the improt settings.
- Fixed an error in the SailPos component when the Sail param was empty

Version 1.07

- Made some changes to the sail impact system, it can now use Box Colliders.
- Square Sail prefabs been updated to have box colliders, tri sails use a simple convex proxy
- Sail ripple system replaced with a ripple texture, makes the shader faster and also needed for the Sail Pos system. If you have sails in the scene which arnt rippling add the Ripple texture to the material
- Added a new component called SailPos, this can be used to position objects on the sail surface.
- Bounds Values added to the Sail component so the sail bounds can easily be adjusted to stop deformed sails being culled early
- Sail Gizmos now includes an orange bounds box with handles for easy adjusting of the bounds area

Version 1.06

- Fixed the help icon on the component inspector bars not having the right link for help
- Added Sail Impacts system for doing damage to your sails

Version 1.05

- Sail param id enum popups are now sorted to make it easier to find the param
- Fixed an issue with the multi edit of sails not updating all the values, works correctly now
- Added helper SetValue methods to the Sail component to be used if updating param values from script SetValue(SailParamID, value);
- Added helper SetValue methods to the SailGroup component to be used if updating param values from script SetValue(SailParamID, value);

Version 1.04

- Fixed an issue with the wind direction and sail movement
- Added a SailSideways param to control how much the sail moves when the wind is edge on
- The Sail Furl component will work in edit mode now for quicker testing of furl order setups

Version 1.03

- Swapped the Voronoi ripple code for Noise based for nicer looking ripples
- Some adjustment to how the FillPercent and SailWind params work, check the docs for new descriptions of the values

Version 1.02

- MultiEdit support added to the Sail component so you can edit multiple sails at the same time
- Fixed sails not updating when play mode entered
- Add a Sail Tilt option to allow you to angle the bottom of the sail relative to the top
- Add a Top arch option to the shader
- Add triangular sails objects
- Added option of edges curves for Move Map maker to make more complex move maps in Unity
- Docs updated

Version 1.01

- Added a Standard version of the Sail Shader for builtin
- Improved the Translucency effect

Version 1.0

- Original release

TUTORIAL VIDEOS

Below you will find some tutorial videos which will explain the shader params and

features and some guides on using Full Sail in your projects.

Full Sail - Shader Params



