Slide 1

Slide 2

We'll Set up the blueprints, attach the metasound, and Insert the sound into the level.

We'll create a UI and use this to drive the metasound through blueprints.

Then we'll develop a method of modulating the wind speed randomly as a normal distribution.

This gives us values more likely to be near a mean value and less likely at the extremes, to produce a realistic effect.

Video 1

We will create a folder in the content draw called Blueprint.

Then in that folder we right click and create a Blueprint Class. Th is is an Actor Class so we can insert it in the level, and name it wind effect underscore BP

Next we create the UI. Right click and select User Interface, Widget Blueprint then user widget and name it wind underscore UI

Now select the level and drag in the Wind Effect Blueprint to the level from the Blueprint folder.

Double click the wind effect blueprint to open the editor then from the content draw metasound folder drag the wind metasound to the default scene root so it becomes a child

Compile and save

Go back to the level tab and select the open level blueprint.

Right click in the editor window and select Create a reference to wind effect underscore BP

Video 2

Next we drag a connection from the event begin play node.

Select Create Widget.

Select the class wind underscore UI.

Then select add to viewport.

Join the widget UI return value to the viewport target.

This will display the UI when the program starts

Compile and save....

Now open the Wind underscore UI blueprint.

Select graph.

Create a new variable with type wind effect BP object reference. This will allow the UI to communicate with the main effect blueprint.

Rename the variable – I called it Wind Effect underscore BP Ref

Compile and save

Return to the main level blueprint.

Create a set Wind Effect underscore BP Ref and connect to the viewport.

Connect the wind effect underscore bp we created earlier and connect the UI widget to the target.

This will store the reference to the effect blueprint in the UI blueprint variable we just created.

Compile and save.

Video 3

Return to the wind underscore UI and select designer.

In the Pallette select Panel and drag a control panel onto the designer screen.

Then drag on a Text block for a title.

Ensure the text box is selected and select the top centre anchor point in the inspector.

Change the text in the block to wind effect.

Adjust the width to 200 and select centre justification.

Now position the text box. Change x to –100 and y to 50.

Now we can run the app and you will see the title displayed.

Video 4

Now we will complete the UI in The wind UI designer....

From the Palette Common menu, drag in 3 text blocks and position approximately. These will be the slider labels...

Then drag 3 sliders...

Then 3 more text blocks which will display the numeric value of the slider...

Now we will tidy up the alignment...

Ctrl click the 3 text boxes on the left and in the Details tab change offset left to 120 ....

Now select the 3 sliders and change the offset left to 400 and offset right to 300...

Now select the 3 text boxes on the right and change the left offset to 750. tick size to content and justify text right......

Next we select the first label, top left, and change the content to wind force...

We select the 3 labels and tick size to content....

Select the second label and set the content to Gain...

Select the third label and set the content to pan...

Now select the top row of widgets and set the offset top to 200...

Now select the second row of widgets and set the offset top to 300...

And select the third row of widgets and set the offset top to 400...

Now select the first slider and set the minimum value to 0, the maximum value to 12 and the step to 1...

At the bottom of the details press on value changed

This will add a node in the graph which will trigger when the slider value changes...

Return to the designer and at the top of the details rename the slider – I used WF underscore slider...

Next select the second slider and rename it – I used gain underscore slider...

And rename the final slider pan underscore slider...

Now rename the right side text boxes WF underscore text, gain underscore text and pan underscore text respectively and tick is a variable for each...

Now select the on value changed for the gain slider and set the minimum value to 0 and the maximum value to 1 and the default value to 0.5...

Now select the on value changed for the pan slider and set the minimum value to -1 and the maximum value to 1 and the default value to 0...

For the wind force set the text content to 3.00

Set the gain text content to 0.50...

And the pan text content to 0.00 ...

We will override these values later programmatically...

Video 5

Now we go to the Wind underscore UI graph and begin to code the functionality of the UI...

First we will initialise the UI. Drag a connector from the event construct node and select set show mouse cursor....

Add a get player controller node and join the return value to the target of show mouse cursor. Click io show mouse cursor. This allows the mouse to work in the UI....

In the left panel we can see the variables we created in the designer. Slider variables are created by default and the text box variables we selected.

Video 6

Now we open the wind effect underscore BP blueprint...

We add a custom event node and name it set wind force....

We then add a custom event node and name it set gain....

We then add a custom event node and name it set pan....

These will be triggered from the UI...

Add an integer variable called Wind force...

Add a set wind force node and connect it to the set wind force event node...

Compile and save...

Select the set wind force event node, and in the details panel on the right add an integer input named value...

Connect the new pin to the set wind force node....

Now select set gain event node and add an input float called value...

And similarly for the set pan event node add an input float pin called value...

Video 7

Now we begin to connect the blueprint to the metasound...

Add a set float parameter node from the audio parameter section....

We want the type target is audio component.

We need to use the exact name to reference the metasound so go to the metasound graph and copy the exact name we used for the gain, we called it master gain, all one word with camel case...

Paste this in the in name of the set float parameter...

Connect the set gain float event node and value output to the float parameter node...

Drag in a reference to the Wind metasound and connect to the Set float parameter node target.

We repeat for Pan...

Duplicate the set float parameter node. Connect it to the set pan event, link the wind metasound to the target and the set pan value to the in float...

Copy the exact name of the Pan Input from the wind metasound – in this case pan1...

Paste into the Pan set float parameter In name...

Compile and save...

Video 8

Now we can complete the UI...

For the On value changed Wind force slider we add a set text, text node. This will update the text box with the value of the slider...

We join the value to the in text. The conversion of float to text is added automatically....

Then we drag in the WF underscore text variable and join to the target. This is the reference to the text box...

Then we add the set wind force node from the wind effect blueprint...

Add a reference to wind effect blueprint we created earlier and connect to the target...

Then connect the slider value to the set wind force value. Again, the conversion is done for us...

We repeat this for the Gain and pan sliders...

Add a set text, text node. join to the on value changed node. connect the value to in text.

Add a reference to the text box and connect to the set text target.

Add the Set Gain and set pan custom events and connect to the respective set text nodes.

Connect the wind effect blueprint reference to the target and join the slider value to the value pin...

Compile and save...

Now you can run the level and test the sliders...

I forgot to convert the wind force value to an integer...

Video 9

Now we can return to the wind effect blueprint

First we create the normal distribution random number generator based on the Box Muller algorithm.

Click on add function and name it Box Muller.

We create 2 random floats.

We subtract the first random float from 1.

We take the log base e of this value and multiply by minus 2.

Then take the square root.

We subtract the second random float from 1.

Multiply by 2 times pi.

Take the sine and the cosine of this value.

Multiply both by the by the result from the square root calculated previously.

Go back to the box muller node and add a float output called Rand1.

Then add a float output called Rand2.

Connect these to the 2 multiplication nodes.

Compile and save.

Video 10

Now we return to the wind effect blueprint event graph and add the variables required...

First we add a wind force array to store the mean wind speeds for each wind force value..

We add a float variable called wind force array and set to an array...

Compile and save so we can add the default values...

Add 13 elements and add the mean wind speed values.

These are 0.5, 1.0, 2.0, 4.0, 6.0, 9.0, 12.0, 15.0, 19.0, 22.0, 26.0, 30.0, and 34.0...

Compile and save...

Next, we will add the variables which control the various wind speed components random fluctuations...

First we add rapid, high speed, fluctuation

We create a single float variable called HSWC underscore current speed...

Then a float variable called HSWC underscore period...

This is the time over which the wind speed varies...

Then a float variable called HSWC underscore delta speed...

This is how much the wind speed varies per second...

Then float variables called HSWC underscore mean period and SD period...

Compile and save

Set HSWC underscore mean period and SD period to 0.5 and 0.1 respectively...

Now we add the slow speed wind speed variables controlling the slow random fluctuations...

Like before we add LSWC underscore period, delta speed, current speed, mean period and SD period...

Compile and save

Set LSWC underscore mean period and SD period to 10.0 and 2.0 respectively...

Compile and save

Finally, add a float variable called frame time...

Compile and save...

Video 11

Now we return to the event graph of the wind effect blueprint...

We create a set HSWC function...

First we connect a set HSWC Period node...

Then connect a set HSWC delta speed to it...

Now insert the box muller function...

Get the HSWC mean period and SD period...

Multiply rand1 by the HSWC SD period...

And add the HSWC mean period...

Ensure this value never falls to zero or less by adding a Max node with value 0.1...

Connect the result to the HSWC period pin

Drag in the wind force array and the wind force integer variable.

Select get a reference to an array element and connect the wind force array and the wind force integer... this is the index to the array...

Divide this by 8 and multiply by rand2...

Then add the result of the division...

Divide this by the HSWC period

The result is stored in the HSWC delta speed variable...

Now we copy all of the nodes we created and create a new function called set LSWC...

Paste the nodes into this function and connect to the set LSWC node...

Now we replace the variables...

Drag in get LSWC SD Period and LSWC mean period and replace the HSWC variables...

Change the value in the Max node to 2.0 so this is the minimum period of the slow wind speed fluctuations...

Change the Period variable to LSWC period...

And change the delta speed variable to LSWC delta speed...

Compile and save...

Now I have made a couple of mistakes which I will correct...

Go to the set HSWC function

We need to get the HSWC current wind speed and subtract it from the result we obtained just before the node where we divide by the period...

Similarly go to the LSWC function and do the same with the LSWC current speed...

Compile and save...

Then return to the set HSWC and replace the connection to the addition node with the get array node...

Do the same for the set LSWC function....

Compile and save...

Video 12

Now we complete the main event graph to add the wind force functionality to the metasound...

We add a set HSWC function node and a set LSWC function node to Event Begin play. This initializes the blueprint...

We add a set frame time node and connect it to the event tick. This will be run every game frame.   The frame time is the length of each frame...

We subtract the frame time from the period by getting the HSWC period, subtracting the frame time and then setting the HSWC period...

Check if the period is now less than zero...

Using a branch node connect this condition...

If true set HSWC again, thus creating a new set of random variables...

Now we subtract the frame time from the LSWC period by getting the LSWC period, subtracting the frame time and then setting the LSWC period. I used a separate reference to frame time just to make the graph cleaner...

Again check if the period is now less than zero...

Using a branch node connect this condition...

If true set LSWC again, thus creating a new set of random variables...

Connect the false condition for the first branch to the set LSWC period node...

Now get the HSWC delta speed variable and the frame time variable, and multiply together...

Get the current wind speed and add this to the result...

Add a max function node to ensure the wind speed is always positive and set the HSWC current speed to the new value...

Connect this node to the set HSWC function node and the false condition of the second branch...

Now repeat for the LSWC. Get the LSWC delta speed and multiply with the frame time.

Get the LSWC current speed and add to the result...

Add a max function and set to 0.1 so the LSWC speed never falls below this value...

Connect to a set LSWC current speed node to update the value, and connect this node to the set HSWC current speed node...

We add together the HSWC and LSWC current speeds giving the total wind speed...

We will add a print string node so this total wind speed is displayed when playing in the editor...

This node is connected to the LSWC current speed node, and the total wind speed connected to the in string...

Now we update the wind speed in the metasound.

Drag in the reference to the wind metasound...

Add a set float parameter, target audio component...

Connect the metasound to the target pin...

Connect the computed total wind speed to the in float...

Connect to the print string node...

Go to the metasound and copy the exact name of the wind speed input...

Paste this into the in name...

Compile and save...

Now we can run the level from wind map...

The wind speed values are shown on the left...

Summary

In this video we have created a UI, implemented the wind speed controls, and hooked this up to the metasound...

Putting as much as possible in the game thread blueprint is most efficient and less likely to cause problems...

If necessary add smoothing in the metasound, but keep complex logic away from the metasound, and therefore the audio thread, as much as possible.

Next we will implement the door creak in metasounds. This is the effect at the start and end of this video...

See you next time...