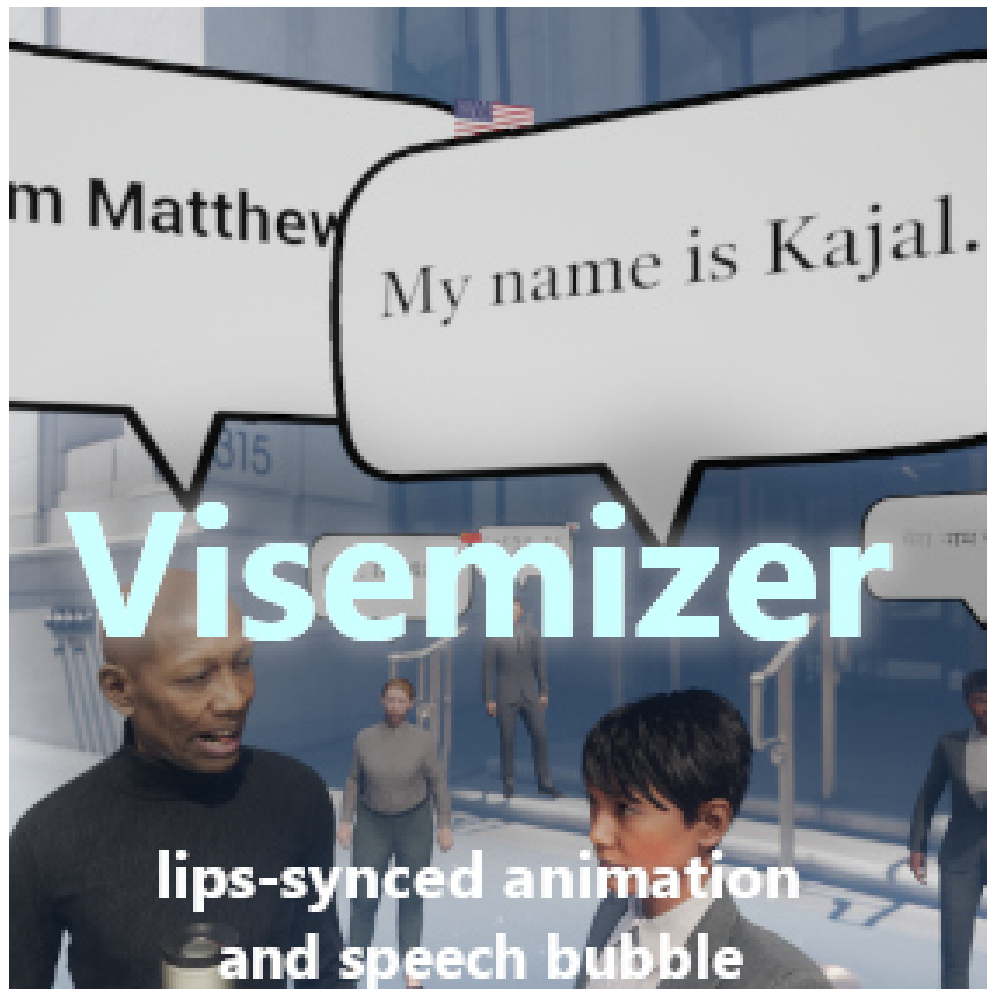


USER GUIDE

Visemizer



by Screenflow

Version 5.5.1, 1st Edition
©2023-2025 Screenflow. All rights reserved.

Made in Germany

This is a documentation belonging to the Visemizer plugin, an EPIC Marketplace product, made for the Unreal Engine 5.3+.

The plugin's sample template and demo are depending on "MetaHumans" skeleton and available as character by MetaHuman Creator. However the product's SpeechMarks Asset import for Amazon Polly can be used independently.

Disclaimer: Amazon Webservices, MetaHumans and Fonts not included. This product does not include support for Amazon Webservices or MetaHumans; please search for documentations of these products.

Generating assets requires an Amazon account during development (not a runtime requirement).

Support for the plugin is available on discord:
<https://discord.gg/Wydr9GVJEt>

Amazon Polly is a trademark of Amazon Web Services EMEA SARL

UNREAL ENGINE, MetaHuman and Quixel Bridge are trademarks of Epic Games, Inc.

Credits: Thanks to Ctker-Free-Vector-Images, OpenClipart-Vectors for their free bubble images on Pixabay.

Release Notes

5.3.2

- *Added Filmmaking / Cinematic Support*

5.3.3

- *Added Templates for MetaHuman Creator.*

5.3.4

- *Rebranding*
- *Interjections Gallery*

5.3.5

- *Editor Tool for Subtitles*
- *Basic Curve Filter for Mood in Layer Blending*

5.4.0

- *Just new engine support. No new features.*

5.5.1

- *City Sample Crowd characters are no longer supported. Demo and documentation updated.*
- *Templates and tools added to support visemes in Paper-2D.*

CONTENTS

PLUGIN INSTALLATION 1

Demo Project Installation 1

Installation in Another (new) Project..... 2

 Use of MetaHuman Creator2

TEXT TO SPEECH AMAZON POLLY 4

Overview 4

Generate Sound 4

 Special Features of Amazon Polly6

Generate Speech Marks 7

UNREAL IMPORT..... 8

UNREAL SUBTITLES..... 10

Accessible Design 10

Subtitle Generator..... 11

UNREAL DEVELOPMENT 12

Getting Started 12

Configure Speech in Outliner 13

Customizing..... 14

Handling Events based on Amazon’s speechmarks	14
UNREAL FILMMAKING	15
UNREAL PAPER 2D.....	16
Sample: Creating Character Viseme Flipbook from MetaHuman.....	17
PLUGIN BLUEPRINT REFERENCE.....	18
Introduction.....	18
AC_Speech	19
Speaking	19
Customization	19
BPI_SpeakingChar.....	21
Widgets.....	21
PLUGIN VISEME	22
Changing or adding viseme.....	22
Changing viseme	23
Adding viseme.....	25
Layer Blending	25
Moods.....	26
PLUGIN INTERJECTION SAMPLE LIBRARY.....	27
Introduction.....	27
Catalog	28

PROJECT VISEMIZER DEMO 29

Overview29
 Legacy versions of plugin and demo with City Sample Crowd .29
Voice/Character Overview30
Character Configuration in Outliner.....31

TROUBLE SHOOTING 32

Possible Issues.....32
 Speech Bubble is black32
 Speech Bubble: My text is missing34
 Speech Bubble: text is overflowing widget34
 Unsupported or incorrect viseme34
 City Sample Crowd does not work.....34
 Arabic or Hindi text does not work34
 Text and video are not in sync.....35
 Speech bubbles in splitscreen do not show face35
 Can I use my own audio (not Polly)?35
 I get an error when I try to reimport speech marks.....35
General Debugging36

PLUGIN INSTALLATION

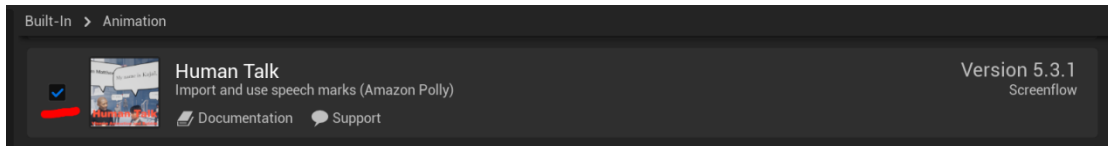
The plugin itself is simply installed over the EPIC games launcher.

DEMO PROJECT INSTALLATION

- *Due to lack of availability, City Sample Crowd characters are no longer supported.*
- *Download and unpack the VisemizerDemo project at your favorite place. You need at least 8 GB free space.*

INSTALLATION IN ANOTHER (NEW) PROJECT

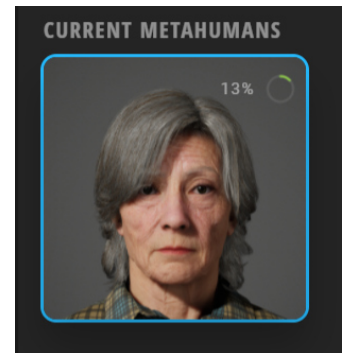
- *Ensure your project is version 5.5 and you have at least 8 GB free space. You may create a new project by launching Unreal Engine 5.5 from the EPIC Games Launcher (we continue this guide with the third person pack).*
- *Open the project and add the plugin. You do not need to restart, just close the project.*



Use of MetaHuman Creator

v5.3.3+

- *Following the EPIC documentation to create and import MetaHumans into your project. In short:*
 - *Go to <https://metahuman.unrealengine.com/> for Meta Human Creator.*
 - *In Unreal Open Quixel Bridge, Login, navigate to "My MetaHumans", download and add them.*
- *Go to the character folder under the MetaHuman folder. Then Drop the character blueprint into a map. Then enable any missing project settings you are prompted for and restart when prompted.*
- *If the character hasn't been dropped in the map drop it again now. You may need to try again, since some operations must complete first.*
- *Press play and check if the character in the level looks okay (see image below).*





- *If everything looks okay, please follow the next steps in this guide.*

TEXT TO SPEECH

AMAZON POLLY

OVERVIEW

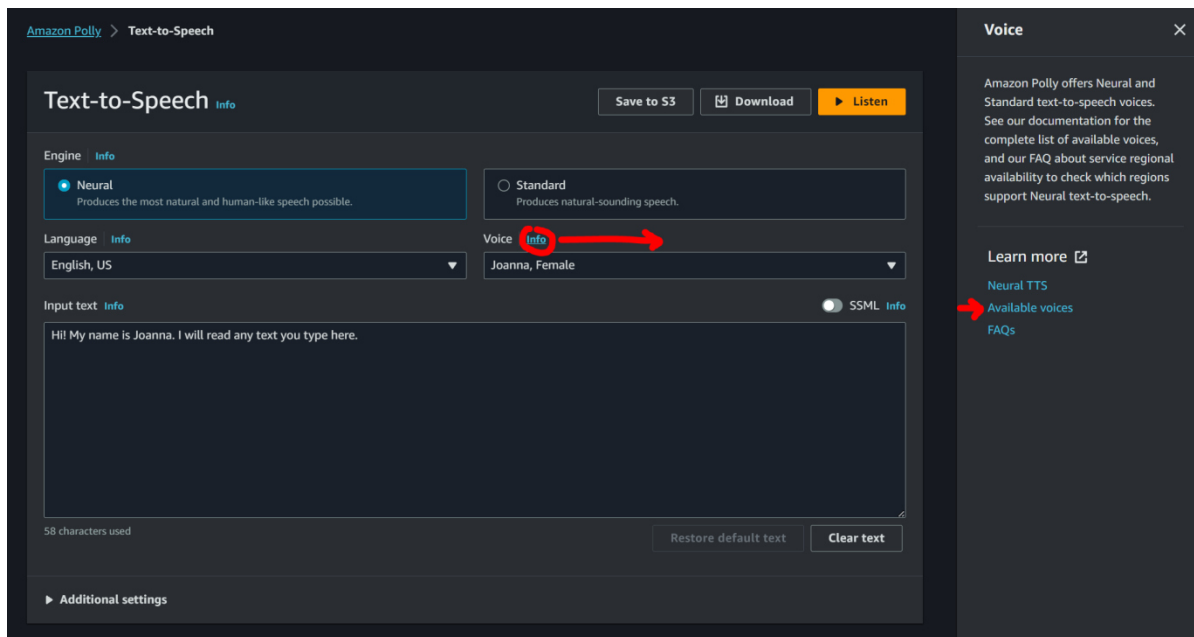
To let the MetaHumans speak, you need to generate the audio and speechmarks from text and import it into your project. This plugin is supporting offline development of speech; but during runtime no account at all is required.

The plugin currently supports only one text-to-speech provider, since it needs speech marks: Amazon Webservices (AWS). You need an AWS account to use the service and you can generate the audio and speech mark files over the browser with it. The following instructions are just a quick overview, consult Amazon's documentation for details. Please be aware that using Amazon services are usually generating moderate extra costs. Some may even be covered in a free tier if you are a starter.

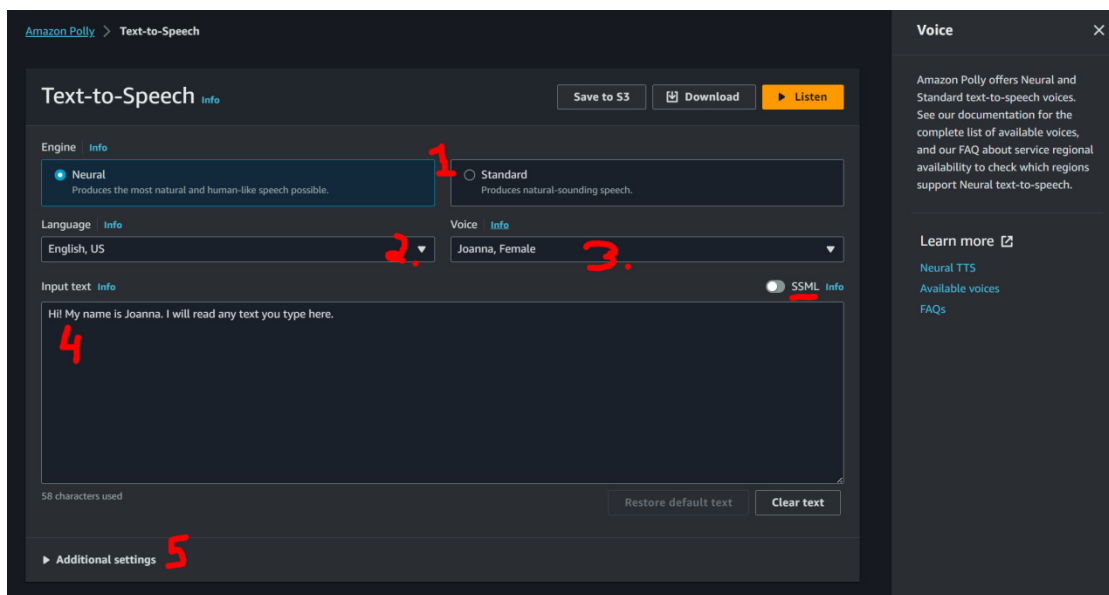
GENERATE SOUND

After you have logged in into the AWS Console, select under "services" and the category "Machine Learning" the service "Amazon Polly". You may want to check if the region in the upper right is the preferred.

You should see the Text-to-Speech screen now. If this is your first time you may want to check for available languages and voices ([click](#)):



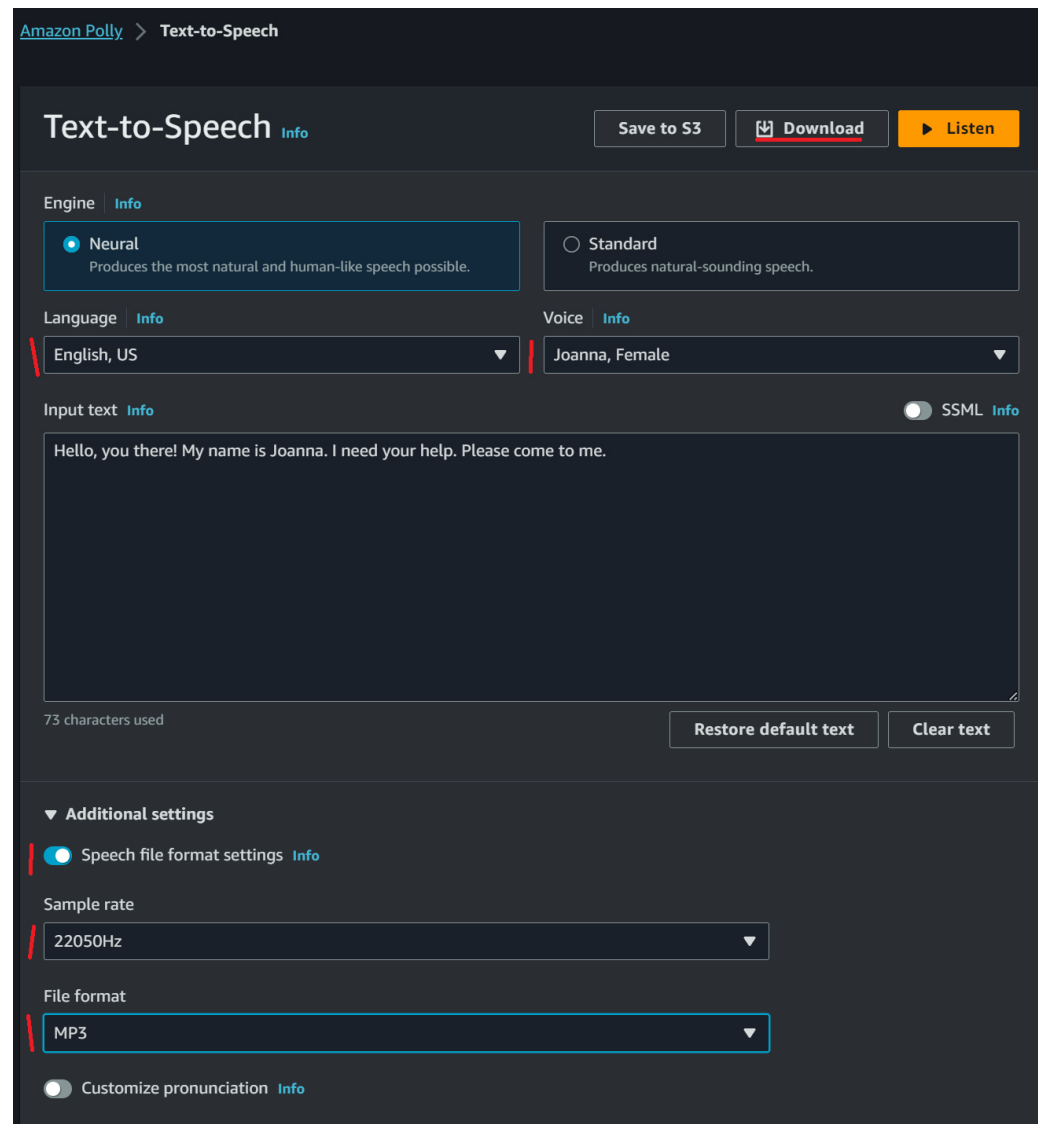
After you have decided what you need, configure the dialogue:



SSML is for advanced user. This will not be covered in this overview, but it is very important for improving your output.

The last step in to configure the advanced settings. We have to download two files, sound and speech marks, so we do this configuration twice before we press the download button each time.

First ensure that speech marks are available for the voice. Then generate the sound file. I recommend mp3 and 22,050kHz.



Special Features of Amazon Polly

Amazon Polly offers some special features that are important to know. Please consult the Amazon documentation about details. Most worth knowing:

- *Adjust Speaking Style, Speech Rate, Pitch, and Loudness*
- *Adjust the Maximum Duration of Speech. This is beneficial for many use cases, especially when it comes to localization.*
- *Amazon Polly Brand Voices (high budget)*
- *Using Phonetic Pronunciation. Tip: <https://en.pons.com/> shows IPA alphabet.*

GENERATE SPEECH MARKS

For the speech marks I recommend the same sample rate. Select all available speech mark types. Include SSML if available.

The screenshot shows the Amazon Polly Text-to-Speech console interface. At the top, there are navigation links for "Amazon Polly" and "Text-to-Speech". The main heading is "Text-to-Speech" with an "Info" link. To the right of the heading are three buttons: "Save to S3", "Download", and "Listen".

The interface is divided into several sections:

- Engine:** Two radio button options: "Neural" (selected) with the description "Produces the most natural and human-like speech possible." and "Standard" with the description "Produces natural-sounding speech."
- Language:** A dropdown menu set to "English, US".
- Voice:** A dropdown menu set to "Joanna, Female".
- Input text:** A text area containing the text "Hello, you there! My name is Joanna. I need your help. Please come to me." Below the text area, it indicates "73 characters used". To the right of the text area is a toggle for "SSML" which is currently turned off. Below the text area are two buttons: "Restore default text" and "Clear text".
- Additional settings:** A section with a dropdown arrow. It contains:
 - A toggle for "Speech file format settings" which is turned on.
 - "Sample rate": A dropdown menu set to "22050Hz".
 - "File format": A dropdown menu set to "Speech Marks".
 - "Speech marks types": A dropdown menu set to "Select speech marks". Below this are three buttons: "Viseme", "Word", and "Sentence", each with an "X" icon to remove it.
 - A toggle for "Customize pronunciation" which is turned off.

Documentation:

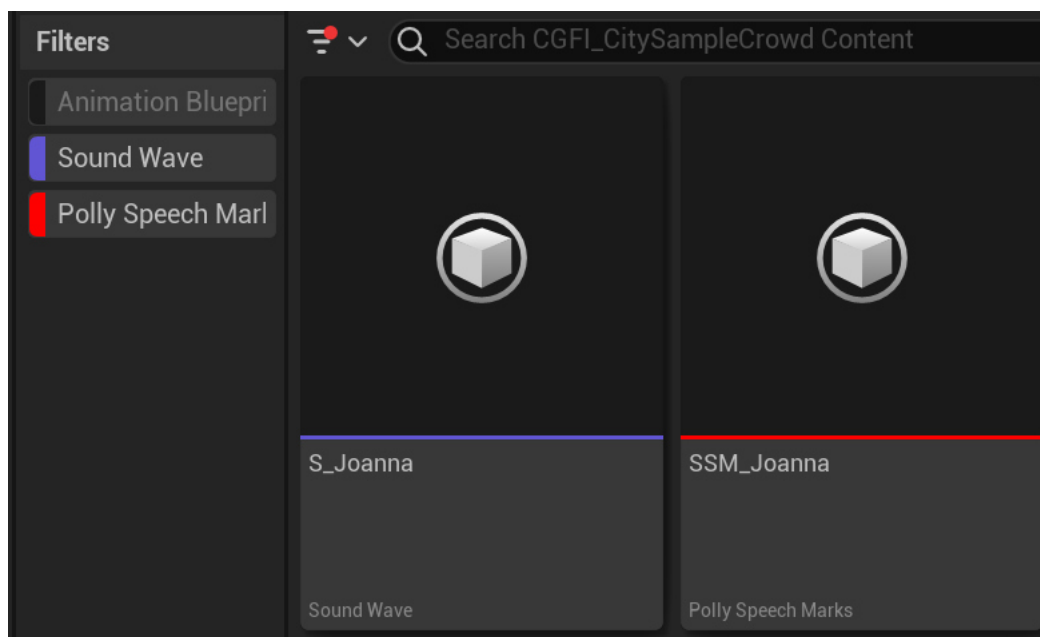
<https://docs.aws.amazon.com/polly/latest/dg/using-speechmarks1.html>

Legacy (before 5.5): After downloading both files, convert mp3 to wav. You can use whatever tool you want. I prefer Audacity[®], but for batch I use ffmpeg, e.g.:

```
ffmpeg -i my.mp3 -acodec pcm_u8 -ar 22050 my.wav
```

UNREAL IMPORT

The import process is simple as usual. Just drag and drop the wav and marks files into your content browser:



The imported sound files are missing attenuation and sound class assignment. You should want to assign at least the sound class, and it best practice to do it within a new sound cue for each sound wave.

The speech marks (SSM) are stored in a UPollySpeechMarks object, that offers access to the speech marks by blueprints. **You have usually nothing to do here, since all is already prepared in a blueprint for you.** If you want, you can look into the speech marks file within Unreal by editing it (don't worry - it is read only):

```

{"time":0,"type":"sentence","start":7,"end":91,"value":"Falsches Üben von \"Xylophonmusik\" <!--break time=\"1s\"--> quält je  

{"time":6,"type":"word","start":7,"end":15,"value":"Falsches")  

{"time":6,"type":"viseme","value":"f")  

{"time":150,"type":"viseme","value":"a")  

{"time":228,"type":"viseme","value":"t")  

{"time":279,"type":"viseme","value":"s")  

{"time":374,"type":"viseme","value":"@")  

{"time":427,"type":"viseme","value":"s")  

{"time":517,"type":"word","start":16,"end":21,"value":"Üben")  

{"time":517,"type":"viseme","value":"k")  

{"time":576,"type":"viseme","value":"u")  

{"time":657,"type":"viseme","value":"p")  

{"time":708,"type":"viseme","value":"@")  

{"time":744,"type":"viseme","value":"t")  

{"time":814,"type":"word","start":22,"end":25,"value":"von")  

{"time":814,"type":"viseme","value":"f")  

{"time":919,"type":"viseme","value":"o")  

{"time":1003,"type":"viseme","value":"t")  

{"time":1068,"type":"word","start":27,"end":40,"value":"Xylophonmusik")  

{"time":1068,"type":"viseme","value":"k")  

{"time":1144,"type":"viseme","value":"s")  

{"time":1216,"type":"viseme","value":"u")  

{"time":1265,"type":"viseme","value":"t")  

{"time":1333,"type":"viseme","value":"o")  

{"time":1410,"type":"viseme","value":"f")  

{"time":1566,"type":"viseme","value":"o")  

{"time":1655,"type":"viseme","value":"t")  

{"time":1721,"type":"viseme","value":"p")  

{"time":1793,"type":"viseme","value":"u")  

{"time":1859,"type":"viseme","value":"s")  

{"time":1968,"type":"viseme","value":"t")  

{"time":2101,"type":"viseme","value":"k")  

{"time":2238,"type":"viseme","value":"sil")  

{"time":2335,"type":"word","start":42,"end":60,"value":"<!--break time=\"1s\"-->")  

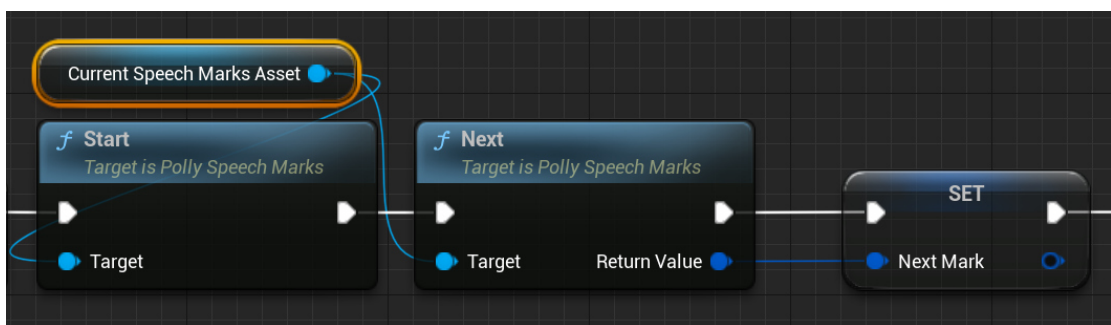
{"time":3438,"type":"word","start":61,"end":67,"value":"quält")  

{"time":3438,"type":"viseme","value":"k")  

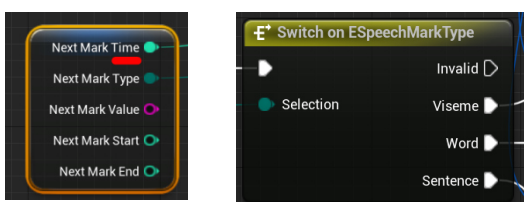
{"time":3507,"type":"viseme","value":"f")

```

In case you indeed want to access PollySpeechMarks yourself, please study the blueprint AC_Speech: every time a speech is started 2 functions need to be called.



Keep track of the time stamps; call Next again when the time passes. The meaning of the speech mark depends on it's type.



UNREAL

SUBTITLES ^{v5.3.5+}

ACCESSIBLE DESIGN

Unreal Engine offers the option to enhance the experience for disabilities by common accessibility features to help. Subtitles is one of them and it is enabled by default in the project settings. However, subtitles are not displayed until sound wave assets are configured with them.

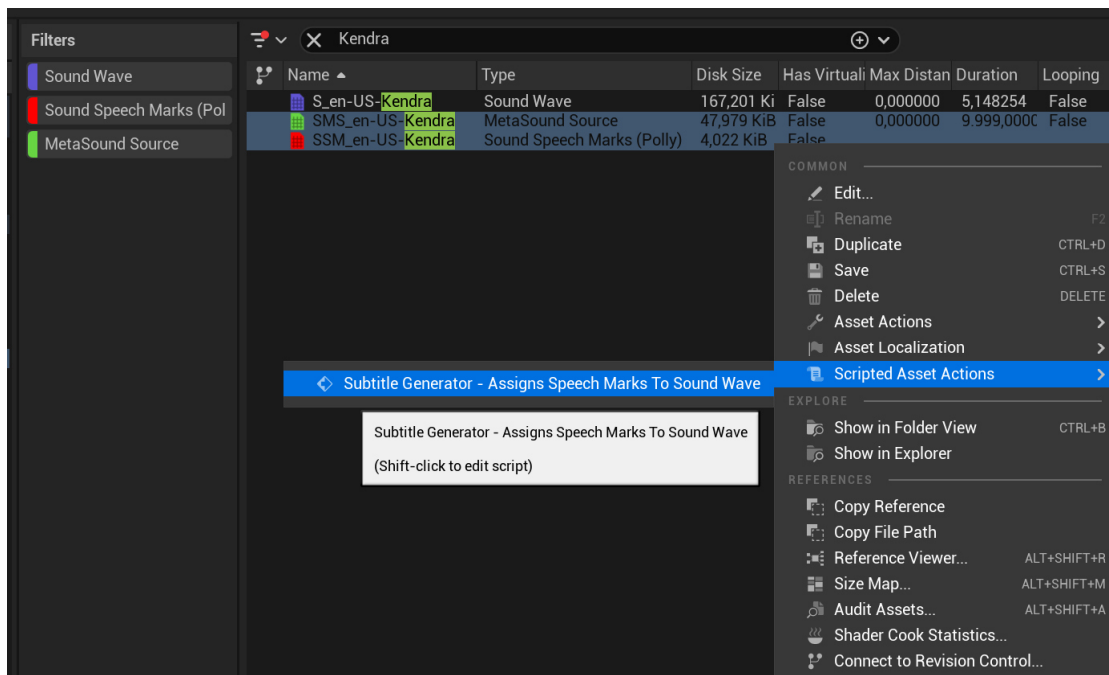
The plugin contains an editor tool that allows you to assign the text from the speech marks to the sound waves, to speed up this process for you. Read the next section how to do this.

The demo has also the subtitle font configured (project settings), you may want to change this as well for your project.

SUBTITLE GENERATOR

The Plugin contains an editor asset blueprint that allows you to copy the subtitles from the speech marks asset to the sound wave asset. It will overwrite any existing subtitles.

Select one speech mark and one sound wave (legacy or MetaSound Source), then right-click on them to open the context menu and select, under Scripted Asset Actions, the Subtitle Generator:



Check the Output Log for any errors.

As result the subtitle texts and times are copied the sound wave and the asset automatically saved. The Priority and Rating (Mature) must be adjusted manually.



UNREAL

DEVELOPMENT

GETTING STARTED

After you followed the instructions above, you should have now a project with the Human Talk plugin, Meta Human content and some speech assets imported. As example we assume you are in the DemoMap based on the ThirdPersonMap.

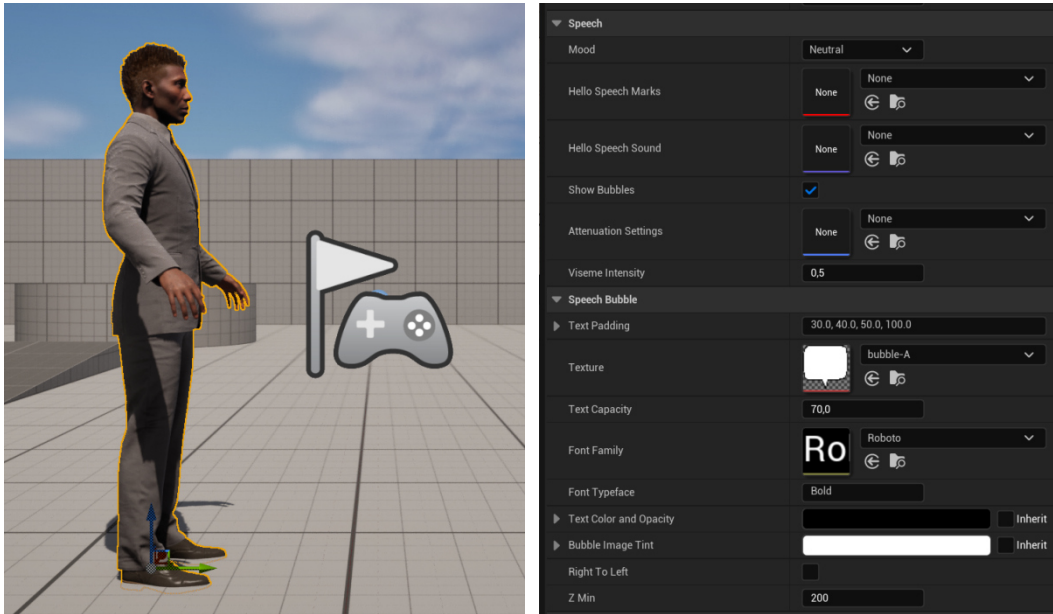
Go to the Content Browser, ensure Setting "Show Plugin Content" is enabled, and navigate to the Blueprint BP_SpeakingMHParent in the Visemizer (HumanTalk) Content under Templates/MetaHuman/Character. This is the new parent blueprint of the actor blueprint that comes with the MetaHuman you have previously imported.

Copy the parent into your project folder, copy the MetaHuman blueprint as well and open the copy, then:

1. *Reparent the MetaHuman blueprint.*
2. *Change Face Animation to AnimationBlueprint ABP_SpeakingMetaHuman_Head*
3. *Tag Face component with face (for exactly attached sound)*
4. *Add call to parent function at ConstructionScript and BeginPlay*
5. *Break link after "On Anim Initialized (Face)", since it may interact with speech and impact animation performance in editor*

CONFIGURE SPEECH IN OUTLINER

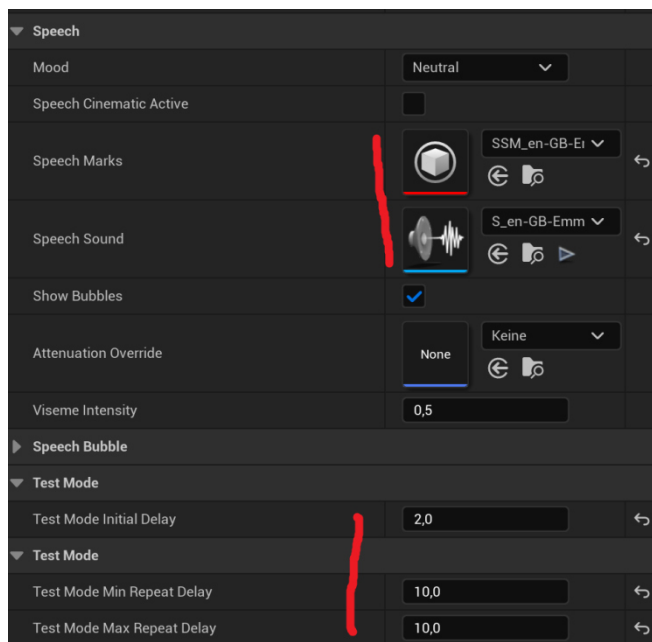
Check the speech sections:



As you can see the speech marks and sound assets, we have imported before, can be assigned here. The other options are explained in more detail at the blueprint reference section below.

You will find sample bubble images at the Templates/Bubbles folder.

With just a couple of changes, the character will speak:

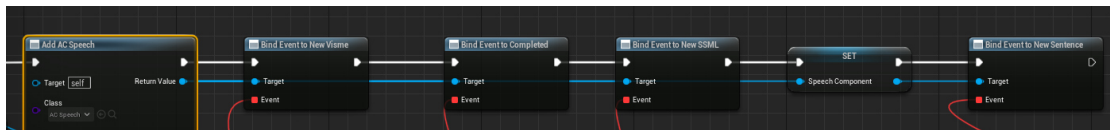


CUSTOMIZING

The plugin allows you to customize the AC_Speech Actor component when needed. But in some cases, you can just use the events it delivers and need not to change it directly.

Handling Events based on Amazon's speechmarks

You can subscribe to four events:



NewViseme: This is called for every viseme. Here you should forward the value to the animation blueprint over the `BPI_SpeakingChar` interface.

Completed: This is called when the end of the speechmarks is reached. This should be about the same time the audio playback completed.

NewSentence: This is called for sentence-speech mark. The value contains a sentence and you might use it to forward it to your own caption system, e.g. instead of showing the bubbles. (SSML stripped, see below)

NewWord: This is called for word-speech mark. It might be useful in an advanced caption system. (SSML stripped, see below)

NewSSML: SSML speech marks entries are not generated by amazon, even if marked. Instead they add markups tags to word and sentence marks. The plugin tries to detect them, strip them off these marks and send the markup keyword and attributes through this event. As an example the `amazon:breath` SSML is handled by the templates and sent as `breath` viseme to the animation blueprint.

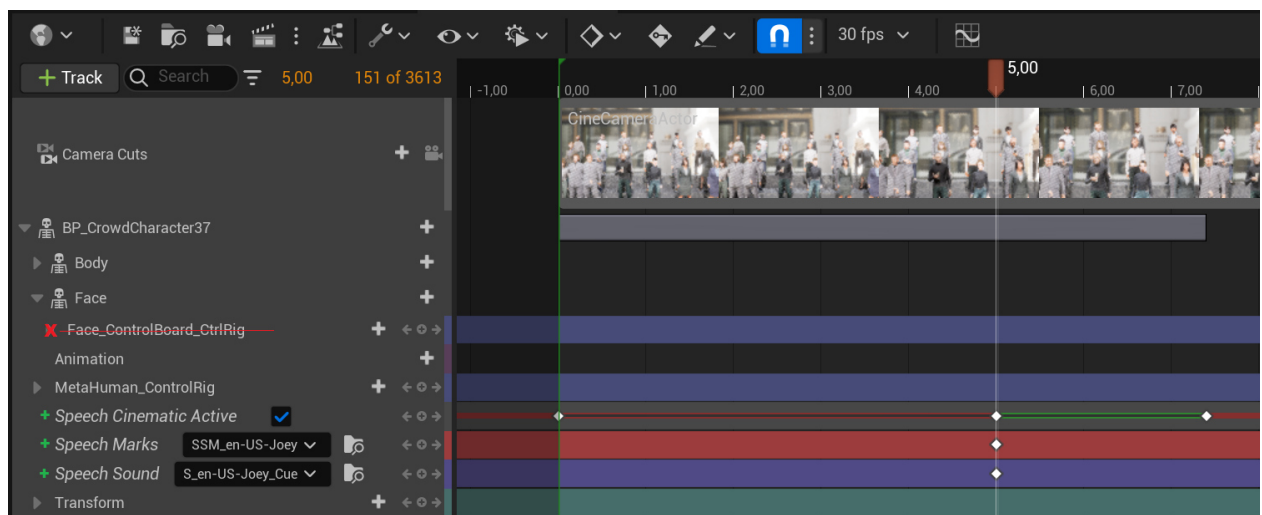
Since Amazon Polly is an independent service, speech marks content may change anytime. However this does not affect speech marks and waves you have already generated. Future changes, if any, need to be handled in a compatible way.

UNREAL FILMMAKING ^{v5.3.2+}

The character stub and the character of the demo project have been improved to be used in sequencers for cinematics.

After adding the character track, you need to add the speech tracks manually and you need to remove the "Face ControlBoard CtrlRig".

Speech is triggered and stopped by "Cinematic Active".



UNREAL

PAPER 2D^{v5.5+}

Paper 2D is a sprite-based system for creating 2D and 2D/3D hybrid games in Unreal Engine.

To show visemes you must implement and enhance an animation state machine to support flipbooks that represent the current state, based on the current viseme state along with your other states. An introduction can be found at:

<https://dev.epicgames.com/documentation/en-us/unreal-engine/paper-2d-overview-in-unreal-engine>

A sample for this is available in the demo project in level DemoPaper2D

Go to the Content Browser, ensure Setting "Show Plugin Content" is enabled, and navigate to the Blueprint BP_Speaking2DParent in the Visemizer (HumanTalk) Content under Templates/Paper2D/Character. This is the new parent blueprint of your actor's sprite blueprint that has to be developed by yourself.

Copy the parent into your project folder and open your sprite actor blueprint, then:

1. *Reparent the sprite actor blueprint as described above.*
2. *Add call to parent function at ConstructionScript and BeginPlay*



SAMPLE: CREATING CHARACTER VISEME FLIPBOOK FROM METAHUMAN

To speed up building speaking MetaHuman or other speaking character flipbooks, in the demo project content folder "Cinematics" there is a level MH2Flipbook. In there, open the sequence VisemeFlipbookSequenceRoot and produce the movie. It will render pictures of a MetaHuman showing the supported visemes.

In the demo project's folder exist a subfolder "tools" containing a batch script CopyImagesForFlipbook.bat

This is prepared to copy the previously produced images for visemes, you can directly import as textures, and create simplified viseme flipbooks (static animation of just one sprite). You may want to enhance the idle animation of the meta human head and add multiple images to produce a more natural look; you can use this as starting point. Or perhaps you might want to produce something different than a MetaHuman, like a speaking comic character.

This examples includes just one viseme to flipbook map for the happy mood. For other moods, or support more states you have to extend the system.

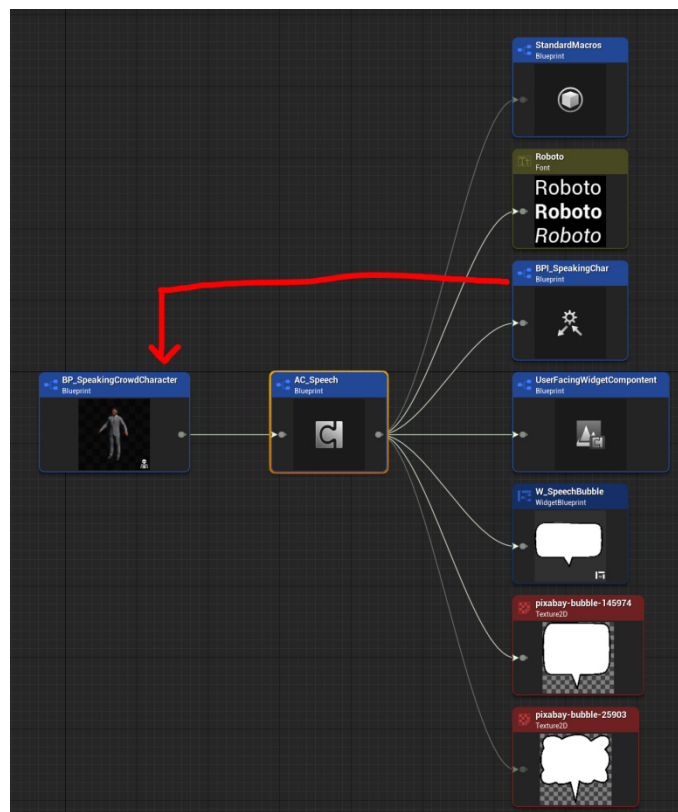
PLUGIN BLUEPRINT REFERENCE

INTRODUCTION

Please look at the dependency graph:

The character (e.g. from demo) requires to add the AC_Speech Component and to implement the interface BPI_SpeakingChar.

You can use whatever character you want, and copy or customize these templates for your needs.



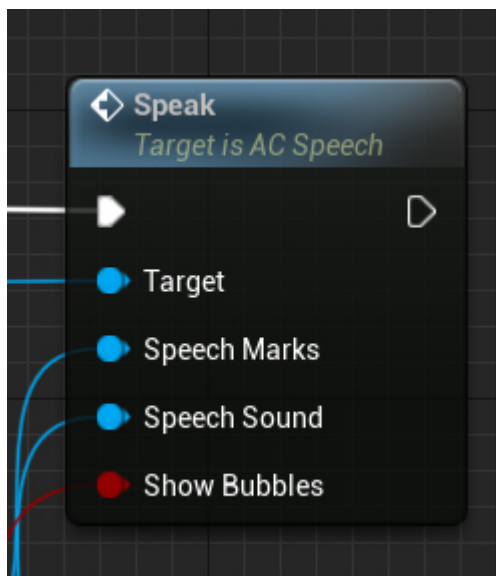
AC_SPEECH

The AC_Speech component contains most of the audio, speech mark and speech bubble handling. It is prepared in a way so you easily adjust the initial setup later in the level outline for you characters, if they support it. Look into the demo character how it is implemented and configured.

The component does not implement replication. This is to the fact, that even in a multiplayer game you may want to keep NPC conversations private to some specific player.

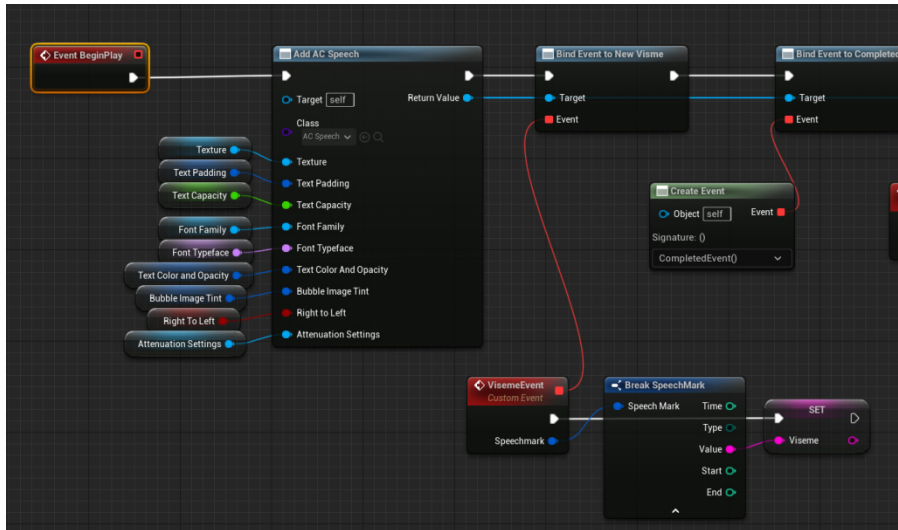
Speaking

To let a character speak, simply call Speak on the component. You have to assign the text-to-speech assets. Showing bubbles is optional; you can still bind to word and sentence events to assign a custom caption system.



Customization

AC_Speech is usually integrated in the following way into your character:



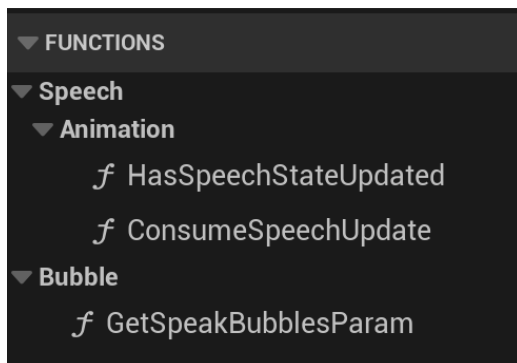
The following values have to be initialized:

Texture	This is the bubble background. Be aware that images should be masked, but (gradually) transparency is not supported. The Plugin contains a small amount of samples from the Pixabay website, but it is recommended that you use your own artwork for your project.
Text Padding	This padding keeps your text with the bounds of your bubble.
Text Capacity	This is a hint how many characters your speech bubble can hold. The templates works per sentence; please ensure your sentences from the text-to-speech provider are not too big.
Font Family Font Typeface	The font of your bubble speech. Some languages are natively supported, some are not. Standard fonts may not be able to display every language. The demo project does not contain fonts for Hindi and Arabic.
Text Color and Opacity	Color of Text
Bubble Image Tint	You can tint the color of your bubble.
Right to Left	This reverses the text. Should be set, e.g. when Arabic.
Attenuation Settings	Option to override any attenuation settings of your sound (cue).

Afterwards you must bind to some events as shown above. At least viseme and complete events must be handled since these are required by the animation.

BPI_SPEAKINGCHAR

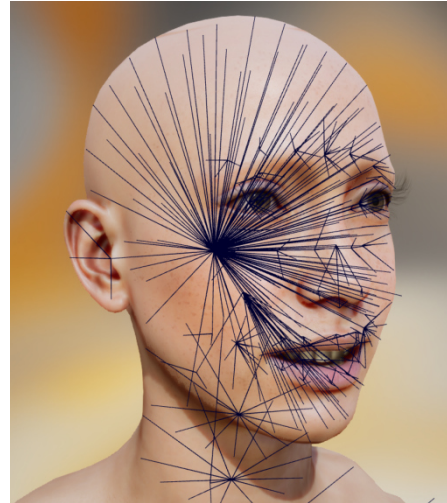
This interface needs to be implemented by the character, so the head animation blueprint template will work correct. The functions are very simple and have a small comment telling what they should do. The parent blueprint templates for MetaHuman and Paper2D already have them implemented.



WIDGETS

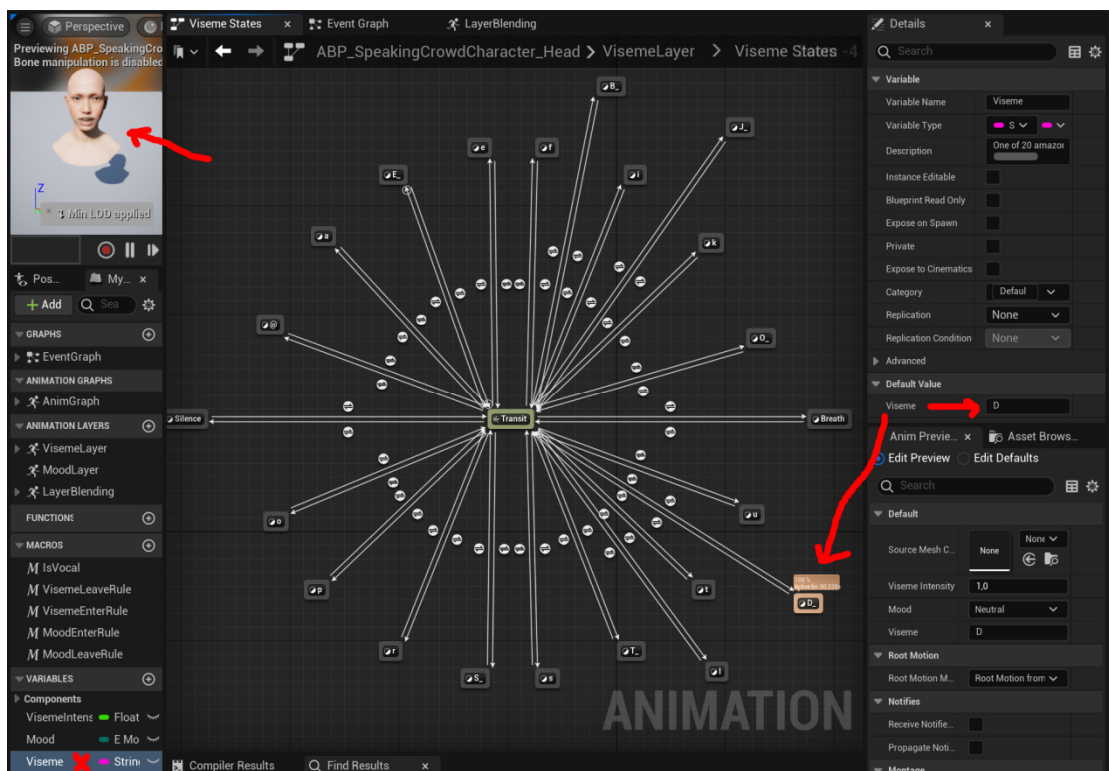
For widgets two templates exist: AC_UserFacingWidgetComponent and W_SpeechBubble. They are very simple and can be used or modified at your demand.

PLUGIN VISEME



CHANGING OR ADDING VISEME

In the Animation Blueprint template `ABP_SpeakingMetaHuman_Head` or `ABP_SpeakingCrowdCharacter_Head` go to the Viseme States:



Then select an existing viseme.

Select the viseme variable and enter the viseme value as defined by Amazon in <https://docs.aws.amazon.com/polly/latest/dg/ref-phoneme-tables-shell.html>.

You will see the graph changing state and the face changing expression. You may need to enable post processing to see it:



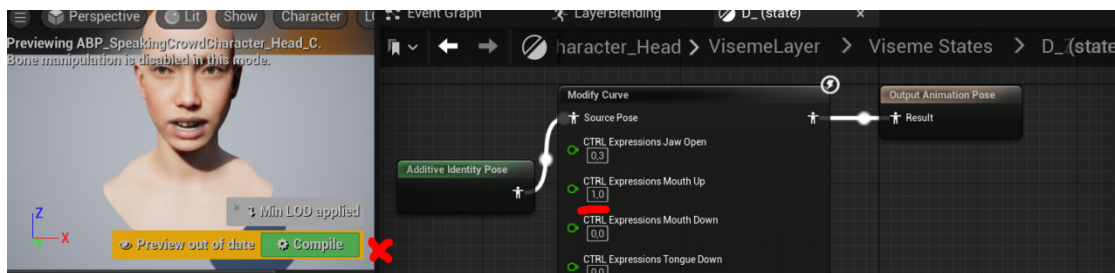
Please note that the state of uppercase letters have a “_” behind their state name, since names can’t be case-sensitive.

Changing viseme

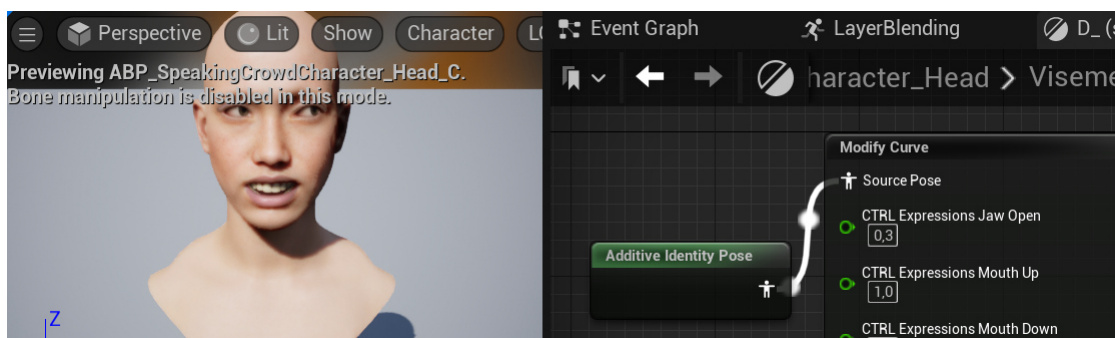
Control Curves used for visemes are coming from the MetaHumans Animator. Please check MetaHumans docs (<https://dev.epicgames.com/documentation/en-us/metahuman/metahuman-facial-description-standard>)

<https://dev.epicgames.com/documentation/en-us/metahuman/metahuman-facial-description-standard>

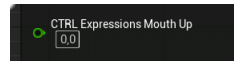
Now go into the viseme state. To understand how to modify it, change the value of Mouth up from 0 to 1. You will see you have to press compile:



You will see the visual expression changes:



Please revert it back to the original value (0) and press compile again.

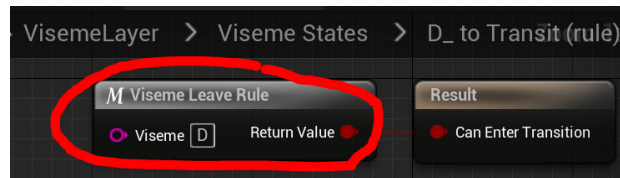
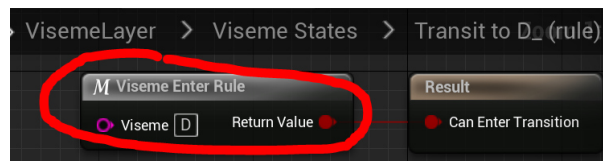


Adding viseme

For now the following Amazon Visemes are supported:

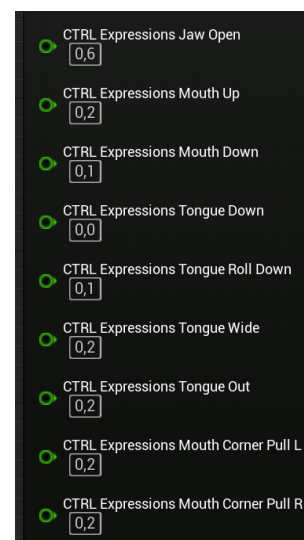
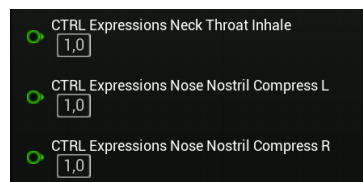
@ a e f i k l o p r s t u B D E J O S T

If an Amazon viseme is not yet supported, or you want to add special visemes and new modes, e.g. for SSML, you have to add a new state. Just copy an existing (one that comes close to it), give it a new name, and add the transitions. Then edit the transit rules and set it up with the existing macros like here:



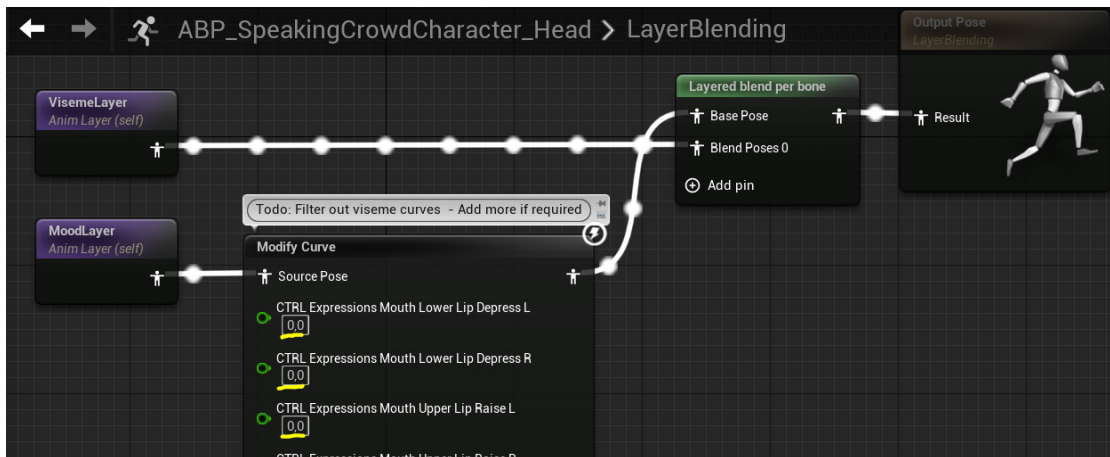
LAYER BLENDING

Blending Visemes with other Animation Layers, like moods and face expressions (e.g. from Live Link), should respect the used curves (numbers vary) and are used by the visemes in the plugin. E.g.:



MOODS

City Sample Crowds had already several mood animations. Unfortunately it is no longer supported. But you can use the old versions to learn how moods could be directly controlled and are combined with the visemes. Change the value of the property mood in the character template.



The Layered blend per bone is (by default) configured with the curve blend option "Use Max Value". Add curves to a filter like above to the mood layer to let them being controlled by the viseme layer.

PLUGIN

INTERJECTION

SAMPLE LIBRARY

INTRODUCTION

The plugin contains a small library of ready-to-use interjection sounds with speech marks, pre-generated with Amazon Polly for your projects. Perhaps they are useful as quick start for some common player emotes.

This is just a tiny selection of interjections for general audience used by English people. As a suggestion what else you might look for:

https://en.wiktionary.org/wiki/Category:Interjections_by_language

CATALOG

The sounds are located below the plugin content folder under "interjections". Prefix of the sound files is S_, they should be embedded in a MetaSound Source, before used. Speech Marks have the suffix SSM_. The interjections have been produced for three different American voices per gender: Danielle, Salli, Joanna, Gregory, Matthew and Joey.

List of included interjections (English):

Interjection	Polly Text (SSML)
Hi!	<speak>Hi!</speak>
Hello!	<speak>Hello!</speak>
Bye!	<speak>Bye!</speak>
Psst...	<speak><prosody volume="x-soft" rate="60%"><phoneme alphabet="ipa" ph=" pssst">psst...</phoneme></prosody></speak>
Yes!	<speak>Yes!</speak>
No!	<speak>No!</speak>
Thanks!	<speak>Thanks!</speak>
Move!	<speak><prosody volume="loud" rate="70%"><amazon:effect name="drc">Move!</amazon:effect></prosody></speak>
Help!	<speak><prosody volume="x-loud" rate="70%"><amazon:effect name="drc">Help!</amazon:effect></prosody></speak>
Enroute.	<speak><prosody rate="85%">Enroute.</prosody></speak>
There!	<speak><prosody rate="145%">There!</prosody></speak>
Gotcha!	<speak><prosody rate="145%">Gotcha!</prosody></speak>
Stop!	<speak><prosody volume="x-loud" rate="140%"><amazon:effect name="drc">Stop!</amazon:effect></prosody></speak>

Note: You should adjust the "viseme intensity" value at the speech blueprint according to the volume of the interjection. E.g.: 0 for Psst and 1 for Stop.

PROJECT VISEMIZER DEMO



OVERVIEW

The sample project (without plugin) is available for download here:

<https://screenflow-support.s3.eu-central-1.amazonaws.com/public/HumanTalk/Visemizer-Demo-5.5.1.zip>

Legacy versions of plugin and demo with City Sample Crowd

Use of City Sample Crown is no longer supported. This section is for tutorial purposes.

The old sample projects (without plugin) are available for download here:

<https://screenflow-support.s3.eu-central-1.amazonaws.com/public/HumanTalk/Visemizer-Demo-5.4.2.zip>

<https://screenflow-support.s3.eu-central-1.amazonaws.com/public/HumanTalk/Visemizer-Demo-5.3.5.zip>

Please follow the installation chapter to add City Sample Crowds before you open it!

When you play the editor default map you will see that characters are speaking one after each other their Amazon default text, including their voice name. Some you may not be able to read, unless you are a language guru.

Bubble images with flags are just for demo purposes.



The demo map is a modification of City Sample Crowds. Characters of the original map have been replaced by a child blueprint from the demo. You can use this child blueprint as starting point for your own work.

VOICE/CHARACTER OVERVIEW

3	Hiujin	Zhiyu	Takumi	Kazuha	Kajal(F)	Zayd(F)
2	Vicky	Daniel	Hannah	Liam	Rémi	Léa
1	Kendra(S)	Joey	Matthew	Emma	Kajal	Joanna(N)

(N) - Newsspeaker

(F) - Sample requires Font for Bubbles not included in demo.

(C) – Actor has been placed in sequencer of this demo

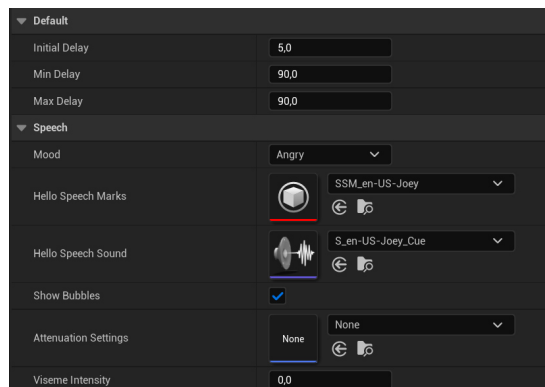
Each character can be configured in the Level Outliner. Choose Joey, with the number 26:



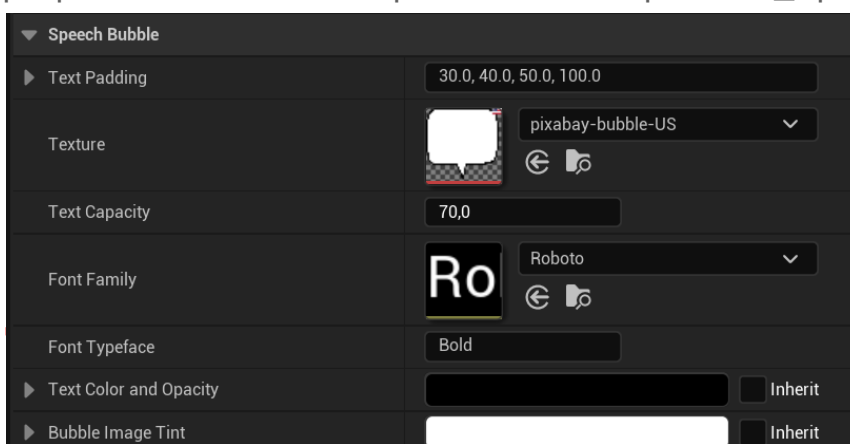
CHARACTER CONFIGURATION IN OUTLINER

The City Sample Crowds Characters are great to configure the appearance. But the child blueprint has some new properties. First some that control the demo talk. Of course you will later use a programmatic approach to let the character speak, according to your conversation rules, but let's use this for a quick start this. Most of the values are explained for the blueprint AC_Speech in the chapter above.

The delay values are used to control that the characters speak after each other (at least for one demo run), speech is controlling what to say.



To control the speech bubbles this section is available. Again many properties have been explained for blueprint AC_Speech.



The value ZMin is referring to the character size.

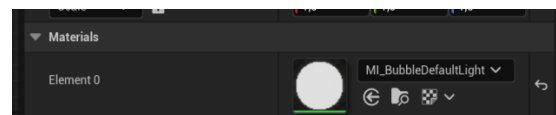
TROUBLE SHOOTING

POSSIBLE ISSUES

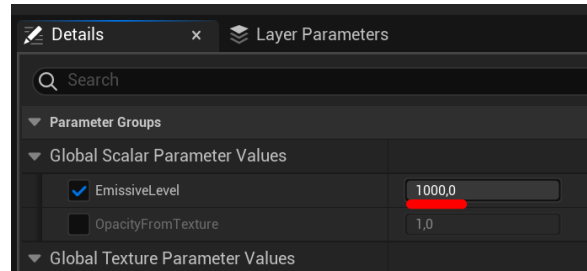
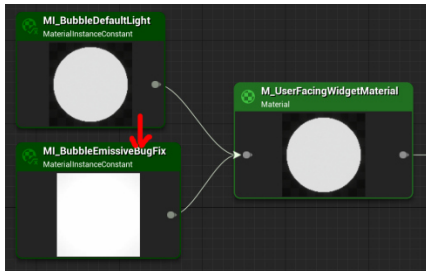
Speech Bubble is black

This can happen if scene illumination is too high. To fix this issue you can either change your illumination or change the material of the Widget Component.

The bubble's widget is displayed by the `AC_UserFacingWidgetComponent`. This component uses a Material that you can change.



Replace the Material Instance by the prepared solution, and adjust the EmissiveLevel parameter to match your illumination:



Speech Bubble: My text is missing

Please ensure your sentences from the speech-to-text provider are not too big.

Speech Bubble: text is overflowing widget

Please ensure your sentences from the speech-to-text provider are not too big. Please adjust the AC_Speech configuration for text capacity and text padding. Avoid large sentences with newlines. Use periods (full stop) frequently. Review the speechmark output from the text2speech provider.

Unsupported or incorrect viseme







Amazon Polly will evolve further. If you are not satisfied with existing or missing visemes, you can easily adjust the Animation Blueprint. Check the chapter "Visemes".

City Sample Crowd does not work

It works. The default installation process is just rough. It has been tested for all versions of unreal engine 5. Please follow the install instructions of this manual.

Arabic or Hindi text does not work

You need to download and import fonts into your project. Some OFL fonts can be downloaded at Google. E.g.:

Name ▲	Type
 TiroDevanagariHindi-Italic	Font Face
 TiroDevanagariHindi-Italic_Font	Font
 TiroDevanagariHindi-Regular_Font	Font
 TiroDevanagariHindi-Regular	Font Face
 NotoNaskhArabic_Font	Font
 NotoNaskhArabic	Font Face

Text and video are not in sync

Even if your assets are matching perfect, the animation blueprint does not guarantee that the blueprint is updated every tick, depending on your scalability setting and maybe some engine runtime decisions. You can detect it sometimes at the start if audio is playing but the lips are not moving immediately. You can try to reduce the scalability setting to encounter this effect.

In case you see lips moving, but audio is delayed, you should try MetaSound instead of SoundCue or activate "Prime on Load".

Speech bubbles in splitscreen do not show face

Bubbles for split screens are currently not supported, but you can use any other caption system of your choice to handle them.

Can I use my own audio (not Polly)?

Yes, this is possible! Instead of modifying the blueprint, you just need to manually create the file with .marks suffix. Even it is not standardized and formally documented, it is UTF-8 encoded and has a simple JSON-like syntax:

```
{ "time":0, "type":"sentence", "start":0, "end":3, "value":"Hi!" }
{ "time":12, "type":"word", "start":0, "end":2, "value":"Hi" }
{ "time":12, "type":"viseme", "value":"k" }
{ "time":124, "type":"viseme", "value":"a" }
{ "time":300, "type":"viseme", "value":"sil" }
```

This approach requires you know the visemes of the words using a dictionary, and map the values according to the Amazon documentation (<https://docs.aws.amazon.com/polly/latest/dg/ref-phoneme-tables-shell.html>). This might not be to complex unless you have a lot of text. In later case it is recommended to use Amazon Polly only for speech marks, and adjust the timings to match your audio file.

I get an error when I try to reimport speech marks

Reimport is not supported for speech marks. Delete old assets.

GENERAL DEBUGGING

There are various already predefined boolean switches for debug output in console or screen in the debug sections of the actor or actor component you can use.