

# SDK - ESML

- Basics
- Embodied Speech Tags
  - [View all available animations](#)
  - [Query the animation database](#)
  - [Request a specific animation by name](#)
  - [Request an animation category](#)
  - [Filter categories](#)
  - [Use grouped syntax](#)
  - [Request a custom animation](#)
  - [Use blocking animations](#)
  - [Use bounded animations](#)
  - [Use unbounded animations](#)
  - [Loop animations](#)
  - [Return to a neutral pose](#)
  - [Use specific layers in an animation](#)
- Audio Tags
- TTS Tags
  - [Create a pause in speech](#)
  - [Modify pitch](#)
  - [Modify speech duration](#)
  - [Make Jibo spell out a word](#)
  - [Specify pronunciation](#)
  - [Specify speaking style](#)
- Auto-Tagging
  - [Use autotagging](#)
  - [Customize autotagging in a MIM](#)
  - [Turn off auto-tagging in a direct call on an embodied speech instance](#)
- Auto-Tagging Rules
  - [Standard Rules](#)
  - [Timeline Modifying Rules](#)
- Cheat Sheet
  - [Animations](#)
  - [Emojis](#)
  - [Dances](#)
  - [SSA](#)
  - [SFX](#)
  - [TTS](#)
- Phonemes

## Basics

You provide text that you want to be expressed with character. This text can be plain, or optionally annotated with XML tags (all of which are described in the following section).

Ways to use embodied speech in a skill:

- MIM prompts
- TextToSpeech behaviors and activities
- Library calls

## Embodied Speech Tags

Embodied Speech (es) tags (and its child tags: Animation (anim), Semi-Speech Audio (ssa) and Sound Effects (sfx)) request an animation to be played. An animation can signify many things-- move Jibo's body, play an SSA, play a sound effect sound, play a screen graphic, or a combine of all of these. The vast majority of the animations that are requested via these tags come from the Animation Database (AnimDB). The AnimDB is a central store for animations that Jibo, Inc. create that give the robot character-rich and consistent content. The AnimDB is searchable, which is made possible by various metadata embedded in the animations. This metadata consists of things like: name, layers, categories, and meta terms.

The AnimDB can be found in your skill's node\_modules: `<skill-root>/node_modules/jibo-anim-db-animations/`

Tag Type	Description
ES	You can request anything from here; no default filter is applied.
ANIM	Filter out any tag with <code>ssa-only</code> or <code>sfx-only</code> listed in its meta terms.
SSA	Only include tags with <code>ssa-only</code> listed in its meta terms. SSAs are just audio files wrapped up in a thin layer of animation.
SFX	Only include tags with <code>sfx-only</code> listed in its meta terms. Like SSAs, SFX are just audio files wrapped up in a thin layer of animation.

All examples in this document will use es tags, but nearly everything also applies to child tags (anim, ssa, and sfx). Exceptions will be noted.

SSA and SFX should be thought of as just audio files. As a result, they have a set duration which does not change and thus have less flexibility in timing compared to standard animations.

### View all available animations

1. Run the following commands in terminal to launch the ADA Viewer:  
`git clone git@github.com:CharacterAI/jibo-anim-db-animations.git cd jibo-ar`
2. Go to <http://localhost:3000> to see a sortable list of all animations, categories, and meta-tags.

### Query the animation database

Run `node scripts/query.js` to see all available queries in the database.

### Request a specific animation by name

1. Specify the name attribute in your es tag.

Find a list of animations by using any of the following:

- Opening `<skill-root>/node_modules/jibo-anim-db-animations/animdb.json` in your skill.
- Using the ADA Viewer.
- Querying the ADA.

Official animation names are found in each object's name keys.

Examples:

- `<es name='Eye_Curious_01' />Not sure if trust moose`
- `<es name='Eye_Curious_01'>Not sure if trust moose</es>`

### Request an animation category

1. Specify one or more `cat` attribute in your `es` tag.

Important: We don't support using multiple categories. Weird things happen.

Find a list of categories by using any of the following:

- Opening `<skill-root>/node_modules/jibo-anim-db-animations/animdb.json` in your skill.
- Using the ADA Viewer.
- Querying the ADA.

All available categories are preceded with the `CAT` label in the document.

Examples:

- `<es cat='curious' />Not sure if trust moose`
- `<es cat='confused' />Not sure if trust moose`
- `<es cat='curious'>Not sure if trust moose</es>`

### Filter categories

1. Specify one or more `filter` attributes in your `es` tag.

With the filter attribute, you can specify which meta tags you would like to:

- Include *all* of: No modifier (or `&` modifier in grouped syntax)
  - To retrieve only animations that are meta termed as happy, simply write: `filter='happy'`.
- Include *at least one* of: `?` modifier
  - To retrieve all animations that are meta termed as happy or excited, write: `filter='?happy, ?excited'`.
- Exclude *all* of: `!` modifier
  - To retrieve all animations that are not meta termed as happy, write: `filter='!happy'`.

You can mix and match these as well. For example, you could write: `filter='ssa-only, !happy'`, which would retrieve only animations with `ssa-only` meta terms and that aren't tagged as happy.

It is important to note that categories and meta terms are independent fields in the metadata. It is possible that your filter will yield no matching animations. Furthermore, it is possible to create filters that are impossible to fulfill, e.g. `filter='happy, !happy'`

Examples:

- `<es cat='curious' filter='!ssa-only'>Not sure if trust moose</es>`
- `<es cat='curious' filter='!ssa-only, !happy'>Not sure if trust moose</es>`
- `<es cat='no' filter='nope, ?small, ?medium, !ssa-only, !sfx-only'>Don't mention it`

### Use grouped syntax

In order to help make more complex filter queries more readable, you may use an alternative but equivalent syntax. The following are equivalent to the above:

Examples:

- `<es cat='curious' filter='!(ssa-only)'>Not sure if trust moose</es>`
- `<es cat='curious' filter='!(ssa-only, happy)'>Not sure if trust moose</es>`
- `<es cat='no' filter='&(nope),?(small, medium),!(ssa-only, sfx-only)'>Don't mention it`

All available meta tags are preceded with the `META` label in the `<skill-root>/node_modules/jibo-anim-db-animations/animdb.json` document.

### Request a custom animation

1. If you want to include an animation you've created in your embodied speech, you can use the `path` attribute to specify the animation. The `path` attribute always looks in the skill's top-level `animations` folder rather than the AnimDB.

Important! You cannot use `path` attributes in bounded mode.

See Audio Tags for information on requesting custom audio files.

Examples:

- `<es path='my-moose-anim.keys' />Not sure if trust moose`
- `<es path='my-first-animation.keys' />This is the first animation I made`

### Use blocking animations

When a blocking animation is requested, nothing else will happen while the animation is playing.

1. To use a blocking animation, do not put any text within the `es` tag.

Example (blocking):

- I can twerk. `<es name='Twerk_Right_01' />` There it was.

In contrast, when a non-blocking animation is requested, it will play at the same time as any TTS that follows it. Non-blocking animations can either be bounded (use a specific duration) or unbounded (no specific duration). See below.

### Use bounded animations

With bounded non-blocking animations, the length of the animation will be automatically derived based on what text it surrounds and will match as best it can. Bounded animations should not be used with SSA and SFX tags as it is unlikely you will find a match.

1. To use a bounded, non-blocking animation, put text within the es tag.

Example (bounded non-blocking):

- `<es name='Twerk_Right_01'>I'm twerking right now</es>`

### Use unbounded animations

With unbounded non-blocking animations, the length of the animation will simply be the native duration of the underlying animation.

1. To use an unbounded, non-blocking animation, do not put text within the es tag, and set the `nonBlocking` attribute to `true`.

Example (unbounded non-blocking):

- `<es name='Twerk_Right_01' nonBlocking='true' />I'm twerking right now`

Note: There must be content to the right of an unbounded animation. Unbounded animations without corresponding text will not fire.

For example, the following will not execute: `<es name='Twerk_Right_01' nonBlocking='true' />`

### Loop animations

1. Specify loop behavior via the `loop` attribute in your es tag.

Animations can be requested to have their playback looped in two ways:

1. In conjunction with a bounded animation, set `loop` to `0` and the system will fit as many loops as it can in the time (which can be 0, if there's not enough time for 1).
2. In conjunction with unbounded animations, set `loop` to 1 or greater and the system will play as many loops as specified.

Examples:

- `<es name='Twerk_Right_01' loop='0'>Look at me, I'm twerking right now, for an extended period of time</anim>`
- `<es name='Twerk_Right_01' loop='3' />I just twerked 3 times.`
- `<es name='Twerk_Right_01' loop='3' nonBlocking='true' />I'm twerking right now, 3 times.`

### Return to a neutral pose

By default, when animations complete, Jibo will stay in the final pose (until something else takes over, usually his Attention System). However, you can also specify that Jibo should return his neutral pose when an animation completes.

1. To override this feature so that Jibo ends in a neutral pose, set `endNeutral` to `true` in your es tag.

This can be combined with other es tag features, (such as `nonBlocking`, for example).

Examples:

- `<es name='Twerk_Right_01' endNeutral='true' />I'm twerking right now`
- `<es name='Twerk_Right_01' endNeutral='true' nonBlocking='true' />I'm twerking right now`

### Use specific layers in an animation

1. Specify what Embodied Speech layers to use in the `layers` attribute in your es tag.

Layers in Embodied Speech do not directly correspond to the layers in animation keys files. Each layer in Embodied Speech is known as a `MetaLayer` and can be thought of as a grouping of layers. The `MetaLayers` are as follows:

- Body (which is composed of the following keys layers)
  - Body
- Screen (which is composed of the following keys layers)
  - Eye
  - Eye Texture
  - Overlay

- Overlay Texture
- Pixi
- Background Texture
- Audio (which is composed of the following keys layers)
  - Audio Event

In Embodied Speech, multiple animations can play at the same time if and only if they occupy different MetaLayers. For example: Animation A1 contains only Audio content and animation A2 contains Body and Screen content. A1 and A2 can happily coexist. If, however, A1 was modified to also contain Body content, they would then conflict on the Body layer and one animation would be rejected.

It often the case that you may want to play part of an animation but not others (e.g. you like the Body aspect of an animation but wish it had different Screen content). Similarly, you may wish an animation didn't get blended with some other content (e.g. an animation with only Body content and you don't want Embodied Speech to blend it with other Audio or Screen content via AutoTagging, for example). This is all possible via the layers attribute.

The layers attribute follows the same syntax as the [filter attribute](#), with some notes:

- You cannot mix inclusive and exclusive filtering of layers.
- You can add layers to the animation on the fly with the + label. When adding layers, you add empty layers to the animation so that it blends with other content differently.
- You can use the all keyword to add all the possible animation layers to the animation.

Examples:

- `<es name='Twerk_Right_01' layers='!screen' />Sweet twerk, right?`
- `<es name='Twerk_Right_01' layers='screen' />Sweet twerk, right?`
- `<es name='dance' layers='screen, body' />Sweet twerk, right?`
- `<es name='Twerk_Right_01' layers='+screen' /> Sweet twerk, right?`
- `<es name='Twerk_Right_01' layers='+all' /> Sweet twerk, right?`

## Audio Tags

Audio tags request a custom sound to be played. For built-in sounds from the AnimDB, use ssa or sfx sounds instead.

See corresponding Embodied Speech Tags sections for more information. audio tags works mostly the same way, except that nonBlocking sounds just play from wherever you put them. There is no concept of wrapping text with sound.

Audio files with a specified path will look in to the skill's top-level audiofolder.

Examples:

- `<audio path='animal-sounds/coo.wav' /> Not sure if trust moose`
- `<audio path='/path/to/coo.wav' nonBlocking='true' />Not sure if trust moose`

Supported formats:

- .wav
- .mp3
- .m4a (aka AAC)

## TTS Tags

The Jibo SDK supports SSML-like tags that tell the TTS system how to modify or enhance Jibo's generated speech.

TTS tags of different types can be nested, but TTS tags of the same type will not blend and may result in odd speech.

### Create a pause in speech

1. Use the `break` tag to place a pause in Jibo's speech.
2. Specify the `size` parameter, where `size` is the length of the pause in seconds.

Example:

- Not sure `<break size='0.5' />` if trust moose.

### Modify pitch

1. Use the `pitch` tag to change the pitch of Jibo's voice.
2. Use one of the parameters described below:

param	description
halftone	Add or remove a halftone to the pitch. Positive values add that number of halftones, negative values subtract (down to a pitch value of 0).

param	description
band	Controls the pitch bandwidth. Increasing the value adds more vibrancy in pitch. Decreasing decreases it (try setting it to 0.0).
add	Takes frequency in hertz as a parameter and adds it on top of, or removes from, the pitch (minimum of pitch value of 0).
mult	Takes a value as a parameter and multiplies it on top of the default. Values above 1 will increase the pitch, and values below 1 will decrease it.

Examples:

- `<pitch add="200"> Hi, </pitch><pitch mult="1.2"> there </pitch><break size="1"/> I'm Jibo.`
- `This is a <pitch mult="2.2"> pitch </pitch> test.`
- `<pitch band="1.2"> Hi, there I'm Jibo. </pitch>`
- `<pitch halftone="-5"> This is a pitch test. </pitch>`

#### Modify speech duration

1. Use the `duration` tag to specify how long Jibo should stretch out a word.
2. Use one of the parameters described below:

param	description
stretch	Allows speeding up and slowing down the speech by a multiplier. Values above 1 will increase the duration of the speech, values below will decrease it.
set	Allows setting a specific duration for the enclosed speech to be spoken in seconds. Minimal 25ms - 0.025.

Examples:

- `This is a <duration stretch="3"> duration </duration> test.`
- `<duration stretch="3.0"> This is a duration test. </duration>`

#### Make Jibo spell out a word

1. Use the `say-as spell` tag to have Jibo spell a word instead of say it.

Example:

- `Jibo is spelled <say-as spell="jibo"/>.`

#### Specify pronunciation

1. Use the `phoneme` tag to provide the phonemes for how a word should be pronounced.
2. Specify the `ph` parameter, where `ph` are the phonemes to be spoken.
3. Specify the stress of each vowel by adding the value at the end of the vowel. 0 - No stress, 1 - primary stress, 2 - secondary stress.

Also see the Phonemes section for documentation on the supported phonetic set.

Example:s

- `Does <phoneme ph="b aa n ou"> Bono </phoneme> love moose, too?`
- `Does <phoneme ph="b aa1 n ou0"> Bono </phoneme> love moose, too?`

#### Specify speaking style

1. Specify a variety of speaking styles by using the `style` tag.
2. Use the `set` parameter to specify the style.

Supported styles are: `neutral`, `enthusiastic`, `sheepish`, `confused`, and `confident`.

Example:

- `<style set="enthusiastic"> This is great! </style> But, <style set="confused"> how did they do that? </style>`

## Auto-Tagging

Auto-tagging is a feature of Embodied Speech that will look through what is being said, try to pull out patterns or other special content, and wrap it up in animations and other content automatically.

There are two stages of auto-tagging.

The first takes place early on in the process and are able to modify the timeline of the content to be performed/spoken. These are known as 'timeline modifying' rules. We work hard to ensure that Timeline Modifying rules do not conflict with content that you've requested, but we cannot guarantee that to be true in every case.

The second phase takes place after Embodied Speech has already done its best to insert all content that was requested via tags. These are known as 'standard' rules. Importantly, standard rules are always of lower priority to the animations and sounds that were requested via tags. More succinctly, a standard auto-tagging rule will never take the place of an animation that you requested (unless Embodied Speech was unable to perform what you requested, for whatever reason).

### Use autotagging

1. Simply type text as you'd read it. The Jibo SDK will auto-tag the text automatically.

### Customize autotagging in a MIM

1. Create a new MIM or open an existing one.
2. Click the Add Prompt button.
3. In the prompt, click the Speech Auto-tagging dropdown.
4. Select an option:
  - All On
  - All Off
  - Custom
  - Default
5. If you choose Custom above, click the Edit button that appears to the right of the dropdown.
6. Select the auto-tagging rules to apply to the prompt.
7. Click Done.

### Turn off auto-tagging in a direct call on an embodied speech instance

You can utilize what's known as an `AutoRuleConfig`, which provides fine-grained control over classes of rules.

An `AutoRuleConfig` provides an interface for you to toggle on or off classes of auto-tagging rules. The classes are the following:

Classes	Included Rules
HotWords	HotWords
Punctuation	Question Comma
Voice	EmotionalVoice
Beat	Initiate BeatBlink Blink DoubleBlink

### Auto-Tagging Rules

The following rules are applied automatically to auto-tagged text.

#### Standard Rules

- **Initiate**  
Applied at the beginning of sentences, Jibo will shift his weight in a natural way.
- **Question**  
Applied when a sentence is known to be a question, Jibo will change posture at the end to express he is asking a question.
- **BeatBlink**  
Applied throughout a sentence, based on heuristics around when it makes sense for Jibo to blink while speaking.
- **Blink & DoubleBlink**  
Applied at the end of sentences. Jibo will sometimes blink at the end of period and exclamation sentences, and will DoubleBlink at the end of question sentences.
- **Comma**  
Applied when the text contains text segmented by a comma (and isn't a list).
- **Hot Words**  
Applied when the text contains pre-identified 'Hot Words' or 'Hot Phrases' that we have special content to play for. Most commonly this applied Emojis, but it could be anything.

## Timeline Modifying Rules

- Emotional Voice  
Modifies the TTS voice parameters around sentences based on Jibo's emotional state (This rule does not trample over any pre-existing TTS changes, so if you've already applied some, this rule will be skipped).

## Cheat Sheet

### Animations

```
<anim cat='affection' nonBlocking='true' endNeutral='true' />
<anim cat='confused' nonBlocking='true' endNeutral='true' />
<anim cat='dance' nonBlocking='true' endNeutral='true' />
<anim cat='embarrassed' nonBlocking='true' endNeutral='true' />
<anim cat='excited' nonBlocking='true' endNeutral='true' />
<anim cat='frustrated' nonBlocking='true' endNeutral='true' />
<anim cat='happy' nonBlocking='true' endNeutral='true' />
<anim cat='laughing' nonBlocking='true' endNeutral='true' />
<anim cat='no' nonBlocking='true' endNeutral='true' />
<anim cat='proud' nonBlocking='true' endNeutral='true' />
<anim cat='relieved' nonBlocking='true' endNeutral='true' />
<anim cat='sad' nonBlocking='true' endNeutral='true' />
<anim cat='scared' nonBlocking='true' endNeutral='true' />
<anim cat='surprised' nonBlocking='true' endNeutral='true' />
<anim cat='worried' nonBlocking='true' endNeutral='true' />
<anim cat='yes' nonBlocking='true' endNeutral='true' />
```

### Emojis

```
<anim cat='emoji' filter='!(hf), &(airplane)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(basketball)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(beach)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(car)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(disco-spin)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(football)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(soccer)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(trophy)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(music)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(question-mark)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(star)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(beer)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(cake)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(cheese)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(drumstick)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(coffee)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(fork)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(fish)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(groceries)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(burger)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(hotdog)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(icecream)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(pizza)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(wine)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(christmas-tree)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(fireworks)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(halloween)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(hanukkah)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(thanksgiving)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(clover)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(valentines)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(chocolate)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(bicycle)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(cat)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(laptop)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(dog)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(gift)' nonBlocking='true' />
```

```

<anim cat='emoji' filter='!(hf), &(house)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(laundry)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(lightbulb)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(money)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(popcorn)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(party)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(phone)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(robot)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(sunglasses)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(toilet-paper)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(trash)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(umbrella)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(video-game)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(bird)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(cow)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(earth)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(flower)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(lightning-bolt)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(moon)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(mountain)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(mouse)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(penguin)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(pig)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(bunny)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(rainbow)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(baby)' nonBlocking='true' />
<anim cat='emoji' filter='!(hf), &(heart)' nonBlocking='true' />

```

#### Dances

```

<anim cat='dance' filter='music, rom-upbeat' />
<anim cat='dance' filter='music, rom-ballroom' />
<anim cat='dance' filter='music, rom-silly' />
<anim cat='dance' filter='music, rom-slowdance' />
<anim cat='dance' filter='music, rom-eletronic' />
<anim cat='dance' filter='music, rom-twerk' />
<anim cat='dance' filter='!(music), &(rom-upbeat)' />
<anim cat='dance' filter='!(music), &(rom-ballroom)' />
<anim cat='dance' filter='!(music), &(rom-silly)' />
<anim cat='dance' filter='!(music), &(rom-slowdance)' />
<anim cat='dance' filter='!(music), &(rom-eletronic)' />
<anim cat='dance' filter='!(music), &(rom-twerk)' />

```

#### SSA

```

<ssa cat="proud"/>
<ssa cat="surprised"/>
<ssa cat="confused"/>
<ssa cat="scared"/>
<ssa cat="embarrassed"/>
<ssa cat="affection"/>
<ssa cat="sad"/>
<ssa cat="happy"/>
<ssa cat="disgusted"/>
<ssa cat="yawn"/>
<ssa cat="laughing"/>
<ssa cat="worried"/>
<ssa cat="dontknow"/>
<ssa cat="frustrated"/>
<ssa cat="oops"/>
<ssa cat="question"/>
<ssa cat="thinking"/>
<ssa cat="hello"/>
<ssa cat="goodbye"/>

```



```
<ssa cat="no"/>
<ssa cat="confirm"/>
```

### SFX

```
<sfx cat="bird"/>
<sfx cat="blip"/>
<sfx cat="dog"/>
<sfx cat="drumroll"/>
<sfx cat="egg"/>
<sfx cat="egg"/>
<sfx cat="frying"/>
<sfx cat="heart"/>
<sfx cat="lightbulb"/>
<sfx cat="party"/>
<sfx cat="scanner"/>
<sfx cat="sparkles"/>
<sfx cat="sunshine"/>
<sfx cat="whoosh"/>
```

### TTS

Tags:

```
<break size='X' />
<phoneme ph='xx yy zz'></phoneme>
<pitch halftone="-5"></pitch>
<pitch band="1.2"></pitch>
<pitch add="200"></pitch>
<pitch mult="1.2"></pitch>
<duration stretch="3.0"></duration>
<duration set="1.0"></duration>
<say-as spell="jibo"/>
```

Examples:

```
Not sure <break size='0.5' /> if trust moose
I won't talk for 2 seconds. <break size='2' /> See?
Does <phoneme ph="b aa n ou"> Bono </phoneme> love moose, too?
Does <phoneme ph="b aa1 n ou0"> Bono </phoneme> love moose, too?
<pitch add="200"> Hi, </pitch><pitch mult="1.2">there </pitch><break size="1" />
This is a <pitch mult="2.2"> pitch </pitch> test.
This is a <duration stretch="3"> duration </duration> test.
<duration stretch="3.0"><pitch halftone="-5"> This is a duration and pitch test
<pitch band="1.2"> Hi, there I'm Jibo. </pitch>
Jibo is spelled <say-as spell="jibo" />.
```

### Phonemes

Phone	Example Word	Example Phonetic Spelling
a	trap	t r a p
ei	waist	w e i s t
aa	spa	s p aa
ah	comma	k o m ah
b	be	b ii

Phone	Example Word	Example Phonetic Spelling
tʃ	cheese	tʃ i i z
d	dig	d i y g
e	egg	e g
i i	fleece	f l i i s
f	fee	f i i
g	green	g r i i n
h	he	h i i
i y	kit	k i y t
a i	price	p r a i s
dʒ	jab	dʒ a b
k	key	k i i
l	leg	l e g
l f	healed	h i i l f d
l s	cattle	k a dt l s
m	me	m i i
m s	spasm	s p a z m s
n	knee	n i i
n s	garden	g aa r d n s
ŋ	ping	p i y ŋ
oʊ	goat	g oʊ
o	thought	th o t
ɔr	north	n ɔr r th
u	goose	g u s
u o	hood	h u o d
ɔi	choice	tʃ ɔi s
aʊ	mouth	m aʊ th
p	pat	p a t
r	reed	r i i d
s	sea	s i i
ʃ	she	ʃ i i
t	tea	t i i
dt	metal	m e dt l s
θ	thing	θ i y ŋ
ð	that	ð a t
ʊ	strut	s t r ʊ h t
ɝ	nurse	n ɝ r s
v	vat	v a t
w	we	w i i
j	yield	j i i l f d
z	zebra	z i i b r ah
z h	seizure	s i i z h ɝ
SPAU	short pause	SPAU

Phone	Example Word	Example Phonetic Spelling
LPAU	long pause	LPAU
θ, 1, 2	gorilla	g ah1 r iy1 l ahθ