

Avatar Framework

Client API

Reference

Table of Contents

Introduction	3
Function Reference	3
Facial Control Functions	3
setGaze(degrees, duration, [magnitude])	3
Speech Functions	4
sayAudio(name)	4
stopSpeech ()	4
Playback Control Functions	4
freezeToggle ()	4
replay ()	4
setPlayerVolume (level)	5
Event Callback Functions	5
voki_sceneLoaded ()	5
voki_talkStarted ()	5
voki_talkEnded ()	6

Introduction

The Player provides an API (Application Programming Interface) that allows a JavaScript or Flash ActionScript programmer to affect different runtime attributes by making function calls from the hosting HTML page or Flash movie. The API enables communication between the host environment and the embedded Voki Scene.

API functions are available for Scenes embedded in Flash movies and HTML pages. API functions are identical for both JavaScript and ActionScript environments.

Embedding in an HTML page:

To use API functions you must include the 'Full Featured Embed Code' when embedding the Scene in your page. The Full Featured Embed Code is available in the embed wizard.

When calling an API function from within a different frame or window, be sure to reference the page where the JavaScript API Stub Code is embedded.

Note: Basic knowledge of JavaScript is assumed.

Embedding in a Flash movie:

Embed a Scene in your Flash movie by using the 'loadMovie' ActionScript function to load the 'Flash Embed Code'. The 'Flash Embed Code' is available in the embed wizard.

Once you load the Scene into a movie instance, any call to an API function should be formatted as follows:

```
instance.function()
```

where "instance" is the name of the object into which the VHost object was loaded.

Note: Basic knowledge of Flash ActionScript is assumed.

Function Reference

Facial Control Functions

setGaze(degrees, duration, [magnitude])

Set the direction & magnitude of the character head and eye movement.

This call will cause the character to divert the orientation of its gaze to the specified direction, and maintain the new orientation for the specified period of time. The orientation will naturally shift towards the center (default) position when the specified time is up, or when/if the character is requested to speak.

The optional *magnitude* parameter governs the "intensity" of the head & eye movement.

Arguments:

degrees	Required. Numeric. 0-360 (0 deg.=top, 90 deg.=right, etc.)
duration	Required. Numeric. In Seconds.
magnitude	Optional. Numeric. In percent. 0-100. Default = 100.

Example:

```
setGaze(90, 6)
```

Speech Functions

sayAudio(name)

Play a specific audio by name.

Arguments:

AudioName Required. String. The logical name of the audio as specified within the account.

Example:

```
sayAudio('some audio')
```

stopSpeech ()

Stop the speech of a currently speaking character. If the character is not currently speaking, stopSpeech has no effect (i.e. it does not prevent speech that has not yet begun).

Arguments:

None.

Example:

```
stopSpeech()
```

Playback Control Functions

freezeToggle ()

Toggle between the pause and play states. Pause – character freezes. If the character is speaking, speech is paused. Play – character unfreezes. If the character was previously paused in mid-speech, speech resumes from that point.

Arguments:

None.

Example:

```
freezeToggle()
```

replay ()

Current Scene playback is interrupted, and immediately restarts.

Arguments:

None.

Example:

```
replay()
```

setPlayerVolume (level)

Set playback volume, or mute the audio.

Arguments:

`level` Required. Integer (0-10) – Default = 7.
A value from 0 to 10; 0 is equivalent to mute, 1 is softest, 10 is loudest.

Example:

```
setPlayerVolume(10)
```

Note: Setting the volume to 0, does not stop the speech (lip movement continues) or stop the audio stream. It affects only the volume.

Event Callback Functions

When incorporating a Scene in your web page or Flash application, it can be helpful to know about the status of the Scene. Using event callback functions enables coordination with your page or application.

Event callback functions are supported both in Flash and in HTML pages. The syntax is the same for both.

Embedding in an HTML page:

The Scene calls specific JavaScript functions in your page, if such functions exist. You must **add the appropriate JavaScript functions to your page** in order to receive these calls. Note that you do not need to add any of the callback functions if you do not intend to use them.

Embedding in a Flash movie:

The Scene calls specific ActionScript functions in your movie, if such functions exist. You should **define the callback functions within your movie at the _parent level**. Note that you do not need to add any of the callback functions if you do not intend to use them.

voki_sceneLoaded ()

Triggered when the Scene is fully loaded and displayed, and ready to start speaking. Calls to API functions before Scene has loaded may be ignored or have unexpected effects. Use this callback to verify Scene is ready to accept API calls.

Arguments:

None.

Example:

```
function voki_sceneLoaded(){  
    alert("the scene is loaded");  
}
```

voki_talkStarted ()

Triggered when the character starts talking

Arguments

None.

Example:

```
function voki_talkStarted(){  
}
```

voki_talkEnded ()

Triggered when the character is done talking

Arguments

None.

Example:

```
function voki_talkEnded(){  
}
```