# VO Manager v1.0

Setup and Usage Guide

# Table of Contents

# Importing the Package

There are two ways to import the VO Manager package. You can either:

1. Visit the Unity Asset Store to download and import the package directly inside of Unity or;
2. You can go to GitHub via https://github.com/tvledesignLLC/vo-manager and download one of the packages inside the packages folder.

By default when importing from the Unity Asset Store the package also comes with a demo as well as the default VO Manager package which contains:

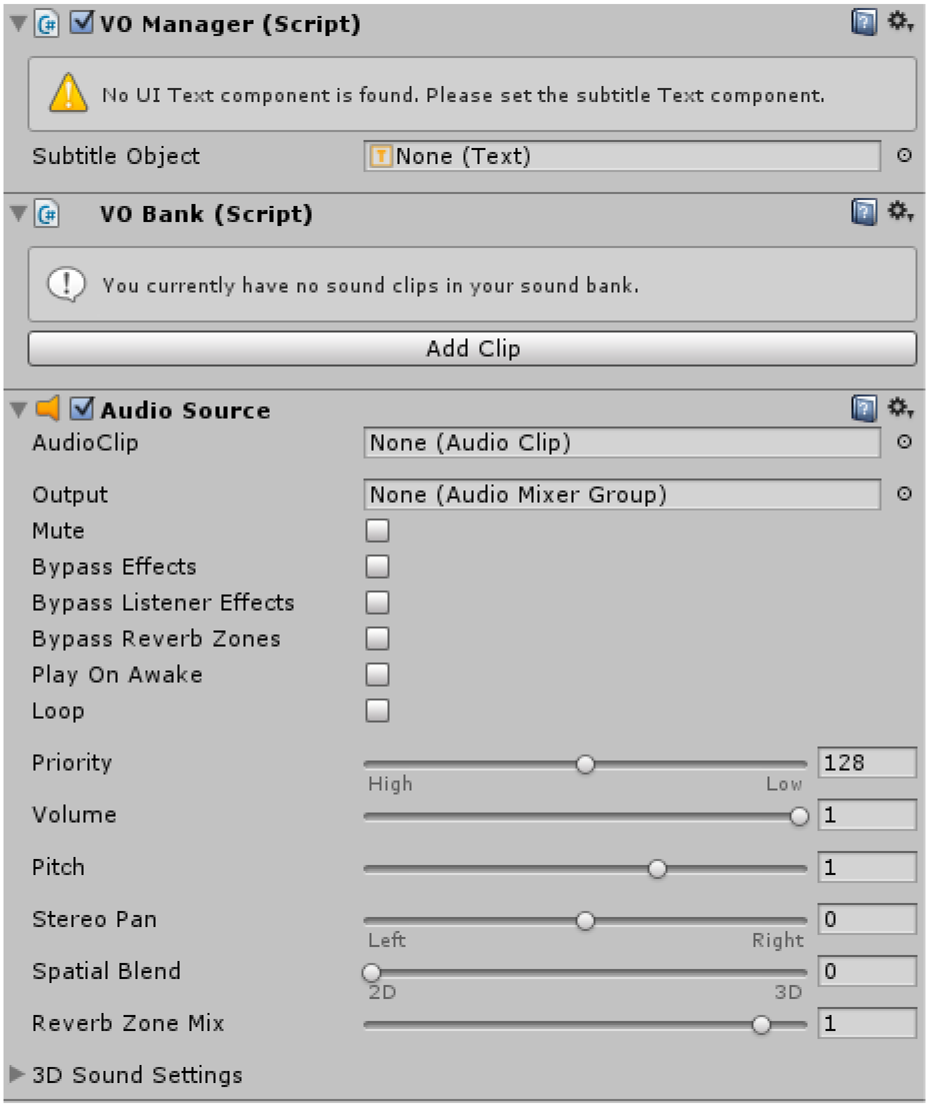
- VOManager
    - Editor
        - VOBankEditor.cs
        - VOManagerEditor.cs
    - Prefabs
        - VOManager.prefab
    - Scripts
        - VOBank.cs
        - VOManager.cs
    - ReadMe.pdf

## Setup

To set up our VOManager you have two of the following options. You can either:

1. Create a new empty GameObject in the Hierarchy and drop the VOBank script or the VOManager script onto it or;
2. You can go into the Project panel and into the Prefabs folder and drag and drop the VOManager prefab into the Hierarchy.

By default when adding the VOBank script or the VOManager script onto a new object an Audio Source component will be added automatically along with one or the other script since they are dependent on each other. You should see a similar interface below when you've successfully setup the initial VOManager.



☑ VO Manager (Script)

No UI Text component is found. Please set the subtitle Text component.

Subtitle Object: None (Text)

VO Bank (Script)

You currently have no sound clips in your sound bank.

Add Clip

☑ Audio Source

AudioClip: None (Audio Clip)

Output: None (Audio Mixer Group)

Mute: □

Bypass Effects: □

Bypass Listener Effects: □

Bypass Reverb Zones: □

Play On Awake: □

Loop: □

Priority: 128 (High – Low)

Volume: 1

Pitch: 1

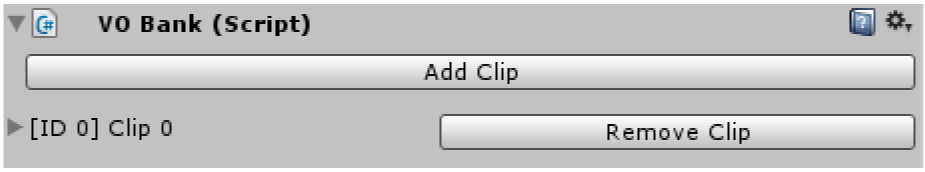
Stereo Pan: 0 (Left – Right)

Spatial Blend: 0 (2D – 3D)

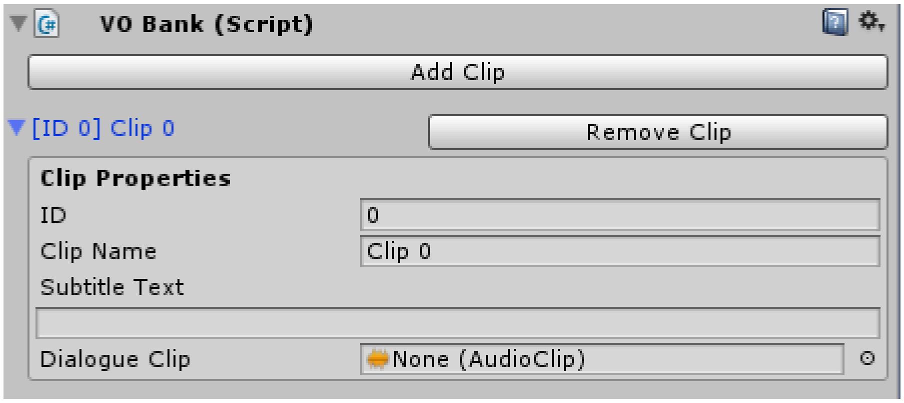
Reverb Zone Mix: 1

3D Sound Settings

Now before we can begin using VOManager we must first start adding VO (voice-over) clips to our VOBank and any subtitle dialogue that we may want to come with it. To do so, click on VOManager in the Hierarchy and press the "Add Clip" button under VOBank. You should see a similar interface below once you click on the add clip button.



Go ahead and click on the expand icon by "[ID 0] Clip 0" to expand the panel.

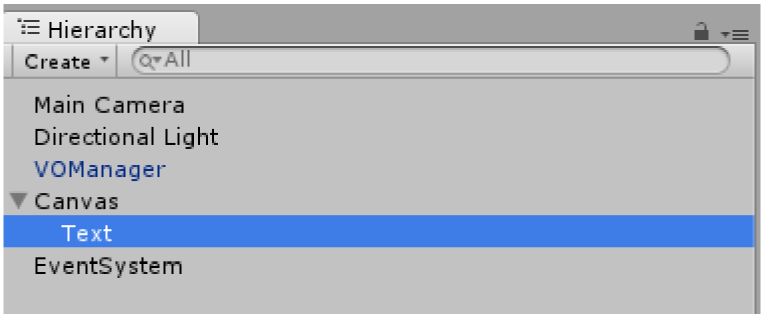


ID: 0
Clip Name: Clip 0
Subtitle Text:
Dialogue Clip: None (AudioClip)

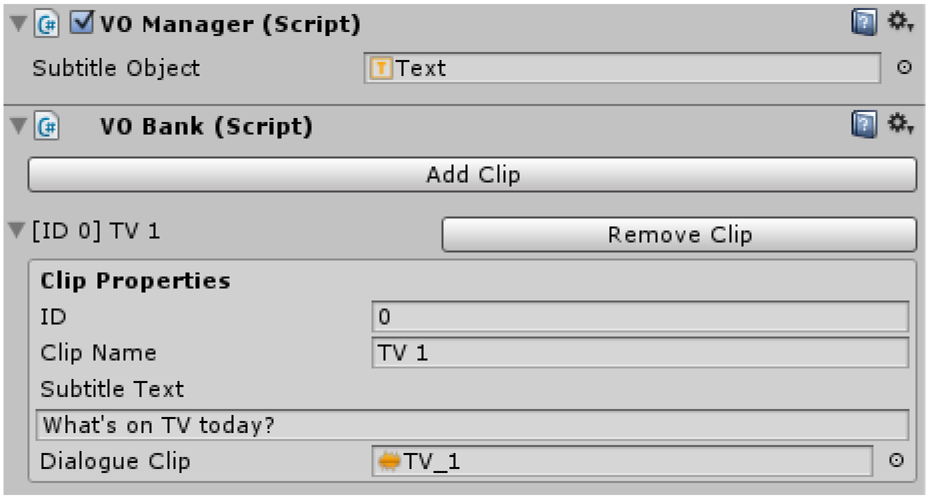
Once expanded you should see a few more options. Here is a list of what they are and do:

- **ID**
  An integer value that auto-increments (non-changeable) based on the position the current clip is in. This is used as a reference to call your clip's properties.
- **Clip Name**
  A string for self-reference. This is mainly use to keep track of all the different clips you may have in the bank for quick reference.
- **Subtitle Text**
  A multi-line text area (string format) that contains the subtitle that you wish to have displayed on the UI when an audio clip is playing (should usually match the dialogue of the clip).
- **Dialogue Clip**
  The audio clip.

You can also remove a clip by simply pressing the "Remove Clip" button. All ID values will reassign itself if you do so. There is however one last step we need to perform before our setup is complete. In order to use our subtitle system we must first have to go to the Hierarchy and create a new Text object. To do this go to Create -> UI -> Text. By default (if you have no Canvas elements in the scene) it will automatically create a Canvas and EventSystem object for you like so:



Setup the Text component properties and values the way you wish to (by default you can keep the text field of the text component empty). Once you are done click back on the VOManager object and drag and drop the Text component we just created (highlighted in blue above) into the **Subtitle Object** field in our VOManager. Once you have at least one sound clip added into your VOBank it should look like the following:



Subtitle Object: Text

ID: 0

Clip Name: TV 1

Subtitle Text: What's on TV today?

Dialogue Clip: TV_1

Once you have finished setting up your VOManager, save your scene and do one of the following:

1. If you used the VOManager prefab you can either Apply the current prefab or;
2. Create a new prefab by dragging your VOManager object into the Project panel (this should be done if you made your VOManager by creating a new GameObject)

## Usage

In order to use our VOManager we must make sure that our VOManager is always present in the scene at all times. We can make calls to our VOManager with the following lines:

- **VOManager.Instance.Play(int id);**
  Plays the dialogue clip normally. Running this function only works if there isn't a current dialogue clip playing.

- **VOManager.Instance.PlayInterrupt(int id);**
  Plays the dialogue clip in "interrupt" mode. Running this function immediately stops the current dialogue clip from playing (if any) and play in the new dialogue clip.

- **VOManager.Instance.Stop();**
  Stops all dialog from playing.

Simply replace "id" with any one of the ID numbers in our bank to play that clip. VOManager by default handles all subtitle text that appears on the screen so you don't have to worry about scripting any kind of functionality. Once a dialogue clip plays the subtitles will automatically appear and disappear once they stop.

## Useful Links

If you have any questions, feedback, or issues, please feel free to contact me via:

- Email at tvledesign@gmail.com or;
- Go to the GitHub Repo at https://github.com/tvledesignLLC/vo-manager

Additionally to learn more or find tutorials you can go to:

- Website at http://www.tvledesign.com
- Blog at http://blog.tvledesign.com
- YouTube Channel at https://www.youtube.com/user/tvledesign

## Special Thanks

Special thanks to these wonderful people who helped contribute to this project:

- Cameron Cintron at http://ccgamedesign.com/ for giving me permission to use the models for the demo scene (which are free to the general public and created during the Global Game Jam 2016 that we took part of)
- Joseph Song for helping with voice overs.

Especially to those who were on my team at the Global Game Jam 2016 which inspired me to create VO Manager:

- Cameron Cintron at http://ccgamedesign.com/
- Jessica Borlovan at http://jmborlovan.com/
- Joseph Song
- Neal Shaw at http://nealryanshaw.com/

Last but not least these wonderful organizations that made this idea possible:

- Global Game Jam at http://globalgamejam.org/
- Petal et Al at http://petaletal.org/