

VR Content Development Guide

Virtual reality content can be developed using a variety of tools. This guide will cover some of the programs commonly used to make content for virtual reality, as well as common problems, design basics and additional considerations.

Engine

Games engines are commonly used to create virtual reality experiences, because they are built to handle rendering and physic interactions, such as collision detection. A developer can also create their own engine, which may run more efficiently by having only necessary functionalities, but engine development is a complex task. Thus, it is recommended that new developers use a pre-built engine.

Unity

One of the most widely used game development engines is Unity. Unity is free to use, thoroughly documented, and many tutorials, asset packages and other resources have been created for it. Unity can be developed using JavaScript or C#, but C# is the generally recommended language.

A basic guide for Unity development is attached in Appendix A.

Unreal Engine

Unreal is a powerful game engine commonly used by large game studios. It has more powerful graphics capabilities than Unity, so it is best to use when creating graphically detailed games. Unreal is free, but has a royalty fee of 5%. The Unreal Engine can be programmed in C++ or using Blueprints, their visual scripting system.

Here are the quick-start guides for VR development in Unreal:

<https://docs.unrealengine.com/latest/INT/Platforms/VR/>

Art

As a visual experience, virtual reality requires art assets. These will generally be 3D models that may or may not have animations. These assets can be either made in a 3D modeling program or acquired online. There are many free 3D models available online.

When creating or acquiring 3D models for VR it is important to:

- Try to use as few polygons as possible – high polygon objects will cause lag
- Limit the number of items; a brick wall should be one object, not a stack of brick models
- Give all standalone objects a thickness – paperthin grass blades and similar objects may render in one eye, but not the other from some angles

Blender

Blender is a free and widely used 3D modeling software. It is a powerful tool, but has a steep learning curve due to requiring the use of keybinds to perform many functions.

Here are the official tutorials for Blender:

<https://www.blender.org/support/tutorials/>

Here is information on importing Blender models into Unity:

<https://docs.unity3d.com/Manual/HOWTO-ImportObjectBlender.html>

SketchUp

SketchUp is a free and simple 3D modeling tool that is very easy to learn. It is not as good for making detailed 3D models and characters, but it is very good at making simple objects quickly.

Here is an introduction guide for SketchUp:

<http://www.sketchup.com/learn/videos/58>

Autodesk Maya

Maya is a 3D modeling and animation program that is extremely powerful and used by most of the top game studios to create models. Maya is an expensive program, but is free for students.

Here is an introduction video to Maya:

<https://www.youtube.com/watch?v=FHNWIECCN3U>

Free 3D Modeling Programs

Other free 3D modeling programs can be found here:

<http://www.hongkiat.com/blog/25-free-3d-modelling-applications-you-should-not-miss/>

Available Assets

There are many places that provide free and premium 3D models that can be imported into your VR experience.

You can find free 3D models here:

<http://opengameart.org/>

You can find many 3D models to use in Unity here:

<https://www.assetstore.unity3d.com/en/>

Audio

Sound is very powerful, especially in virtual reality. Ambient noises, music and sound effects all help to set the tone of the game. A horror game is much less terrifying without sound and sound effects help add weight to interactions in VR – Smashing a door, pressing a button, pouring liquid.

Free Audio

You do not have to create all of your own sounds. There are many websites with free sounds effects and music.

- <http://www.flashkit.com/soundfx/>
- <http://www.audiomicro.com/sound-effects>
- <http://www.freesoundeffects.com/>
- <https://www.freesound.org/>
- <http://soundbible.com/>

Audio Editing Software

There are numerous audio softwares available for recording and editing audio. A commonly used free audio software is Audacity.

Audacity can be found here:

<http://www.audacityteam.org/>

Audio Middleware

Most engines do not have advanced audio controls that are easily accessed and used. Developers often use audio middleware to more easily control and balance audio. The two most commonly used audio middlewares are FMOD and Wwise.

FMOD can be found here:

<http://www.fmod.org/>

Wwise can be found here:

<https://www.audiokinetic.com/products/wwise/>

Design

Developing content for virtual reality is unlike creating anything else. Common practices of film and video games are not always applicable. The increased freedom of the user/viewer means that they can watch anywhere and you have no guarantee that they are ever facing a certain direction. This also applies to how users interact with the environment if you are tracking their hands and is a big part of why simple games that give the player freedom to interact with the objects they see, like *Job Simulator*, have been very popular.

User Interface

User interfaces (UI) have to be completely rethought to work well in virtual reality. While the standard for most games are active UI elements around the edge of the screen and locked position menus, neither of these work well in VR. Your peripheral vision is very sensitive to change, but is unable to focus on elements. Thus, a health bar to the side of the game would be distracting when it changed and impossible to see in detail.

Menus locked in the center of the screen suddenly feel as if they are attached to your head and can feel very uncomfortable. Integrating the UI into the physical environment is the best and most natural solution.

Motion

Motion within a virtual world is the fastest way to send someone reeling. When your eyes witness movement and our inner ear does not feel it, a gut-churning feeling is generally produced. There are some solutions and ways of mitigating the sickness, but, much like car-sickness, not everyone will find VR movement pleasant.

Unsurprisingly, the best way to do VR movement is to simply let the person move themselves around the VR environment. The SteamVR experience was very transformative for many VR enthusiasts, as it allows you to walk around a space instead of being stuck in a chair or a single location. Allowing a person full control of their movement as they naturally would eliminates the motion sickness, as their body experiences the same movement. Movement is still limited by the space of SteamVR.

Placing the player on or in a moving object and keeping the player stationary within the object is one of the best solutions to motion troubles. Games like Vox Machinae and Eve: Valkyrie place you in command of a mech or space ship, allowing you to feel perfectly snug in your chair while controlling the movement of your craft. For most people, having stationary elements around them is all that is needed to solve the movement problems in VR.

With our current technology, sometimes you'll want to sit in a chair and explore the world in a first person virtual view. There are a few easy ways to make this as painless as possible. First, the movement speed should be kept fairly low. Oculus recommends 1.4m/s for a walk and 3m/s for a jog. Movement above this speed starts to make one feel pretty nauseous. Second, acceleration is much more disorienting than constant velocities. It is much better to have movement that goes from 0 to speed rather than have a noticeable acceleration at the start. This is particularly important for turning.

Finally, it can be very disorienting for "forward" to not be the direction you are looking, so consider making forward movement based on the camera direction. While it might be hard to have good movement controls that only allow you to move in the direction you are looking, it is smart to avoid gameplay that requires looking in one direction while moving in another, especially when the player does not have full control of that movement.

Camera Control

VR experiences can be in first or third person. Sometimes VR experiences are best presented laid out in front of you, like a board game on a table. The most important aspect of camera use in VR, though, is that the control of the camera always belongs to the player. Many non-VR games take the camera away from the player to pan across a new landscape or to set up a better view angle for a dialog scene. These camera motions done in VR bring about the same nausea that poor movement causes for similar reasons, because camera movement in VR is interpreted as motion by your brain.

Best Practices

Oculus created a best practices document for developing virtual reality experiences, which can be found here:

https://developer3.oculus.com/documentation/intro-vr/latest/concepts/bp_intro/

Appendix A: Unity Basics and Resources

Unity Projects

Unity projects consist of the project file, assets and scenes (files that end in .unity). Scenes are the files and spaces within which content is shown and developed. To open a Unity project once it has been created, you must load a scene. When you save a Unity project, be sure to save both the scene and the project itself.

When a project is ready to be shared, you can build the project by going to File>Build Settings, picking the appropriate settings for the platform you want to run the program on, and hitting “Build”.

Assets

Assets are components of an experience such as 3D models, textures, and sounds. Assets created outside of Unity must be saved in a file format that Unity can read and then imported into Unity.

This is a guide for importing assets into Unity:

<https://docs.unity3d.com/Manual/ImportingAssets.html>

An important asset type for many VR experiences is a Skybox. A Skybox is rendered around the whole scene and acts like a sky. This is necessary for any VR experience that will be in an “outside” space or in an interior space with windows.

The basics of Skyboxes can be found here:

<https://docs.unity3d.com/Manual/class-Skybox.html>

Virtual Reality Development in Unity

Unity projects require SteamVR to run on virtual reality headsets. SteamVR was developed by Valve to support the creation of virtual reality content in Unity.

You can acquire SteamVR and import it into their Unity project here:

<https://www.assetstore.unity3d.com/en/#!/content/32647>

Once SteamVR has been imported into the Unity project:

- Delete the MainCamera
- Navigate to Assets>SteamVR>Prefabs and drag and drop the [CameraRig] into the scene
- If you are not using tracked controllers, you can delete Controller (left) and Controller (right) from the [CameraRig]

A basic tutorial for working with SteamVR can be found here:

<https://www.youtube.com/watch?v=LZTctk19sx8>

The API documentation for SteamVR can be found here:

<https://github.com/ValveSoftware/openvr/wiki/API-Documentation>

A troubleshooting guide for SteamVR and the Vive can be found here:

https://support.steampowered.com/kb_article.php?ref=8566-SDZC-9326

Additional help and information can be found here:

<https://steamcommunity.com/steamvr>

Tutorials and Resources

Since Unity is so widely used, searching Google for trouble you are experiencing or questions you have should generally produce a guide or solution. If you cannot find a solution, ask on the Unity forums or Stack Overflow.

The official Unity forums can be found here:

<https://forum.unity3d.com/>

The official Unity tutorials can be found here:

<https://unity3d.com/learn/tutorials>