The Thesis Committee for Jared Jamesen LeClaire
Certifies that this is the approved version of the following thesis:

Virtual Reality Collaboration for the Entertainment Industry

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor: ______________________________

James Glavan

Co-**Supervisor:** ______________________________

Sven Ortel

Virtual Reality Collaboration for the Entertainment Industry

by

Jared Jamesen LeClaire, B.A.

Thesis

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Fine Arts

The University of Texas at Austin

May 2016

Abstract

Virtual Reality Collaboration for the Entertainment Industry

Jared Jamesen LeClaire, M.F.A
The University of Texas at Austin, 2016

Supervisor: James Glavan
Co-supervisor: Sven Ortel

This project is an exploration of emerging virtual reality and gaming technology as a means of collaboration for creative teams in live performance, film, and themed entertainment. As computer generated graphics technology continues to advance I believe it is necessary for artists to explore the creative possibilities these technologies offer and push the boundaries to influence continued development. This thesis asks questions of the effectiveness of VR and gaming technology as a collaborative tool, how these technologies can work in conjunction with traditional design processes, and details my experiences working with this technology on a devised theatrical production.

Table of Contents

List of Figures

Chapter 1: Introduction

For the modern artists who work in entertainment, computer graphics and virtual reality technology can make what was once impossible, possible. Virtual reality can offer huge advantages to the creative teams for any theatrical production, film, or other aspect of the entertainment industry. This technology allows designers, directors, and technical staff to collaborate on all aspects of a production in an extremely careful and meticulous manner. Designers can spend months working on their respective designs, but rarely does the team have the opportunity to view all design elements of a production as a cohesive whole prior to technical rehearsals. If the design team was provided the opportunity to immerse themselves in a complete virtual representation of their design's world where 3D representations of scenic elements, lighting, and costumes could be viewed together in one virtual space, would this offer any real advantages to the collaborative process?

It has always been my instinct to leverage technology for the purposes of creativity. It is an instinct that inspired me to pursue projection design as a career in the first place. Currently, the primary consumer market for virtual reality is for video games, but there has been steadily growing interest in other fields as well, such as architecture, medical fields, and the military. I intend to explore its value as a collaborative tool for entertainment designers. I'm drawn to exploring virtual reality because, much like projection design, it is still a medium in its infancy and its continued relevancy will depend on innovative individuals finding new and exciting ways to use the technology.

In recent years, many different pieces of creative software have become available as tools for entertainment designers. Scenic designers can use programs such as Vectorworks and AutoCAD to draft drawings. Lighting designers can use WYSIWYG, ESP Vision, or Spotlight to lighting visualizations. Costume designers can use Photoshop to digitally paint renderings. Sound designers can record and mix audio with ProTools or Logic. Projection designers can mock-up stage looks in d3, Watchout, or Pandoras Box. But still, each of these tools are encapsulated and generally incapable of intercommunication in the manner I'm looking for. For instance, if a lighting designer I am collaborating with does their renderings in WYSIWYG and I'm working in d3, there is no intuitive way to merge the two files and view them as a whole. The creative team is forced to put the pieces together in their imagination. I've always desired a single software environment that allowed all the different design disciplines to compile and explore their ideas in real time. I believe the answer may lie in virtual reality and video gaming technology.

Pre-visualization (referred to as 'previz' for the remainder of this document) is a very important aspect of working as a designer in the modern market. The idea of previsualization is not a new idea in the landscape of scenography. Designers have been creating sketches, renderings, and models to communicate their ideas for well over a century now. The difference now is that many of these previz processes are taking place inside of computer software rather than on a drawing board. For instance, ‘d3' is a popular media server and previz environment for video and projection designers. It uses a 3D digital environment to visualize design concepts, help make technical decisions (such as projector/screen placement), and pre-program designs. D3 has an extremely high level of utility but, in my opinion, is often lacking in visual artistry. Many of the visuals that d3 produces often read to me as very cold and stark. While d3 does an exceptional job communicating the technical feasibility of a design, it does very little for the emotional and expressive qualities of an image that often drive creative conversations.

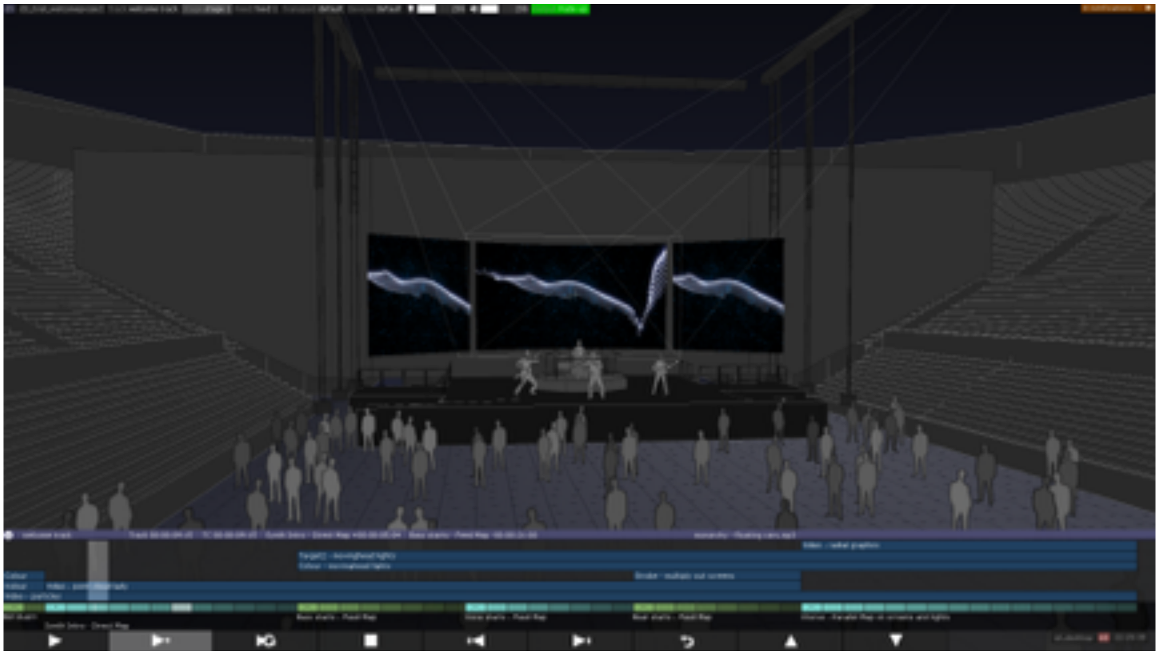

Figure 1: screenshot of the d3 previz environment

Video gaming engines have the graphical capabilities to open up many new and interesting possibilities in the world of previz for designers because they allow for realistic representations of each and every design element inside of an interactive digital environment. In order for an artist to use a tool effectively they must first become familiar with how to use the tool. The first step was to examine the technology involved in integrating video gaming and virtual reality technology into the design process.

Chapter 2: The Technology

Virtual Reality

I first became aware of consumer virtual reality when the Oculus Rift 'Kickstarter' campaign began in 2012. It was always something that I had thought was limited to science fiction, but now it had become a reality. One of the first thoughts I had was 'I wonder how this could be used for design?'. A major focus in my studies as a designer has been how to communicate design ideas in a way that is both artful and honest. I mean 'Artful' in the sense that the images exhibit a great sense of craft and evoke the intended emotional responses from the viewer, and 'Honest' in the sense that what I show sets a reasonable expectation for the quality and appearance of the final product. In order to produce quality digital products with these qualities I had to immerse myself in study of many topics, including virtual reality, computer generated graphics, animation, computing logic, and many more topics. My learning during the course of this project was highly incremental as much of the technology I am working with is cutting edge and does not yet have a regimented method of study. The following sections summarize the process I had to go through in order to create a basic product in virtual reality.

Virtual reality is a new frontier in the digital gadget market. At the time this paper is being written, there are 3 major companies developing virtual reality headsets; Oculus VR (Oculus Rift headset), HTC (Vive headset), and Sony (Morpheus headset). Each headset has its advantages and disadvantages. Oculus Rift was initially funded by an online Kickstarter crowd fund which rewarded backers with a developer kit of their first headset, called the ‘Oculus Rift DK1’. Some time after, they released the Oculus Rift DK2 which was available for consumer purchase. HTC made developer kits for their ‘Vive’ headset, but they were not available for general consumer purchase. Developers had to go through an application process and were selected by HTC. The Sony headset, called ‘Morpheus’, didn’t even exist yet. Due to the ease of access to the Oculus Rift DK2, I decided to use their headset for this project. The Oculus Rift DK2 consists of a head mounted display (HMD), a tracking camera, and a power supply. The HMD consists of an LCD screen mounted in front of two sets of optics. The optics divide the image between the two eyes of the user to help create a stereoscopic 3D effect. All over the headset are small infrared tracking points which are picked up by a USB camera that is mounted above the computer monitor. This tracking data is translated into the game software to simulate actual head movements inside the game.

Figure 2: the Oculus Rift DK2 HMD

Video Game Engines

So why video game engines? As video gaming has surged in popularity over the past years, there have been enormous advancements in the world of real-time computer generated graphics. To sum it up briefly, computers are able to do more faster now. Modern gamers have developed a taste for photorealistic visuals, and as a result hardware and software developers have been striving for more efficient ways to generate realistic lighting, textures, audio, and physics within games. Video games are also specifically designed to be interacted with by people and offer many tools for creating interactive environments. The gaming engine is the software environment in which these games are created. The software provides a host of tools relevant to creating video games, such as modeling, material, and lighting tools. These qualities of modern gaming engines make them an ideal environment for design.

Before beginning my search for the right gaming engine, I identified the initial key features that I wanted from the software:

- Virtual reality support (specifically the Oculus Rift DK2)
- The ability to freely navigate the virtual space
- The ability to control light (color, beam profile, intensity, patterns, etc.)
- The ability to display and project video textures
- The ability to create realistic textures
- The ability to create realistic aural atmospheres
- The ability to interact with the virtual space

After a little research I quickly discovered that all of these features can be achieved within just about any gaming engine as their explicit purpose is to create an immersive and interactive virtual environment. There are many engines available on the market, such as Unity, CryEngine, Unreal Engine 4, and Source to name a few. After a great deal of research, I ended up choosing to develop with Unreal Engine 4 (referred to as UE4 for the remainder of this document). UE4 has a reputation for high quality lighting and material effects, and is also highly affordable. The primary coding language of UE4 is C++, which is a very common general purpose programming language that is

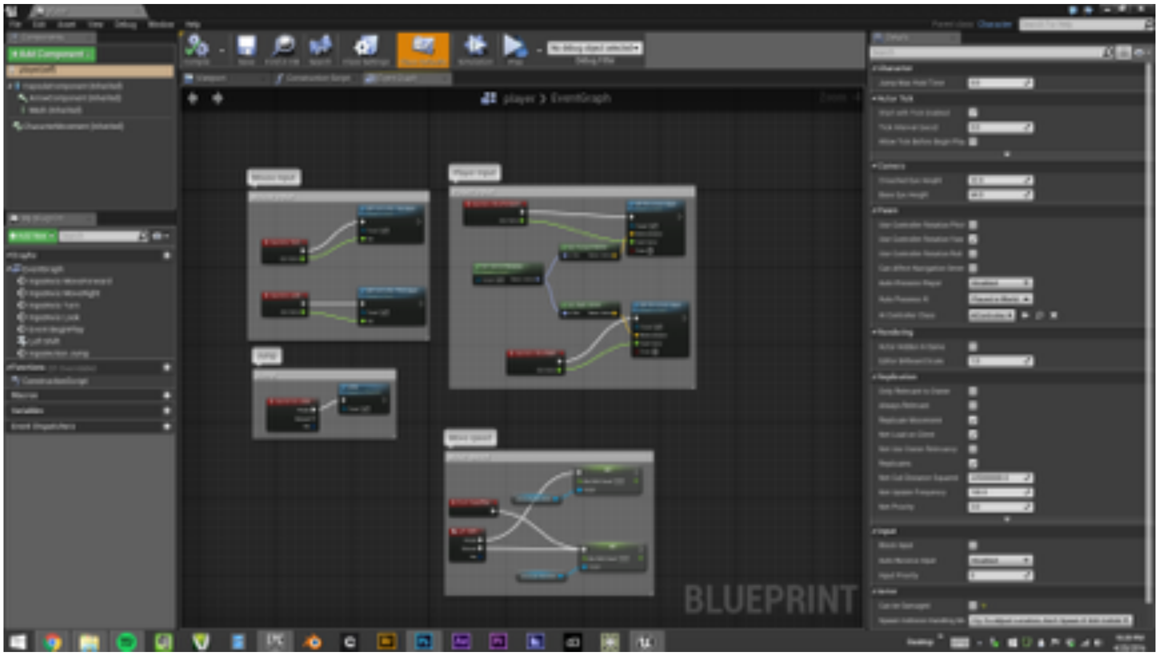

Figure 3: Example of a Blueprint script in UE4

compatible with Windows, Mac, and Linux operating systems. Fortunately, UE4 also uses a visual programming language called 'Blueprints' that can be used to create most scripts necessary for programming simple interaction in the software. This is a huge advantage as I am not experienced with C++ and was not going to be able to learn it within the timeframe of this project. UE4 also has a very active and engaging community that proved to be an invaluable resource while developing the software end of this project.

Workflow

Before putting the software into practice it was necessary for me to develop a workflow that allowed for much efficiency and, most importantly, flexibility. I feel that in order for a collaborative tool to function, both the tool and the user need to be able to react to new ideas in a timely manner. I also wanted to make sure that the workflow was able to integrate traditional design processes as well, because it will be uncommon to find other designers who produce the type of files UE4 requires in order to function. Based on my goals for the use of this tool I identified that I needed to find solutions to the following problems:

1. Creating 3D objects for venues, scenic elements, props, etc. based on drawings, sketches, and drafting of my collaborators
2. Accurately texturing 3D objects
3. How to do the above in a timely manner

Modeling

Fortunately, I entered this project with a history of experience with 3D modeling. I often draft in 3D using Vectorworks and I also have a fair amount of experience with a piece of software called ‘Blender’, which is a free and open source 3D modeling/

animation software. The difficulty comes when I need to model a highly complex or detailed object quickly. After a little research I discovered a technique called 'photogrammetry'. Photogrammetry is a process developed for surveying and mapping that uses photography to determine the distance between objects. During part of this process the software creates a detailed 3D model of the object or landscape being surveyed. If the process is done correctly, it produces an extraordinarily detailed model and texture that can then be processed and placed into a 3D scene. This is can be a very powerful tool for creating models, but there are some disadvantages. If the initial photographs are not taken correctly the software will not be able to generate an accurate model. There are also

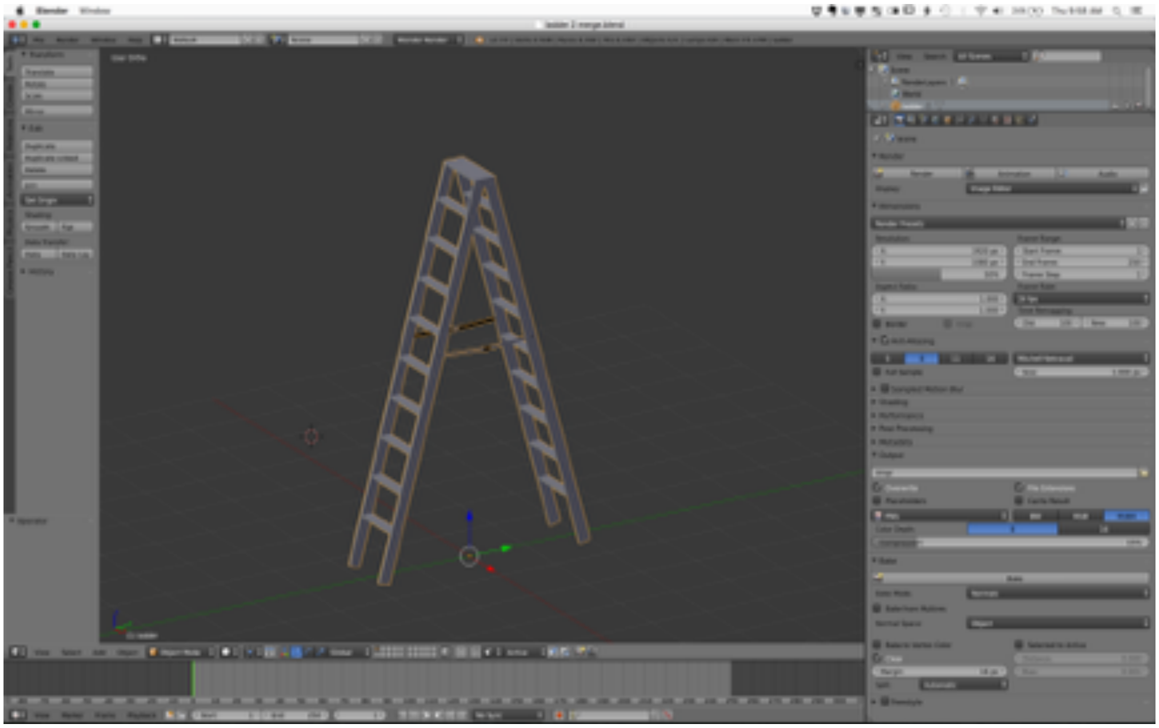

Figure 4: example of Blender interface

many objects that simply can't be scanned, such as objects with very shiny or specular surfaces or objects with repetitive patterns (such as brick or plaid textures).

Texturing

After the model is created it must be textured. This way the object will have color and definition and appear like the real object. If the model was created using photogrammetry there is a high likelihood that the texture is already complete, otherwise, the model will require another step of processing. The first step is to 'unwrap' the model to a 2D raster in order to give all the faces of a model texture coordinates. After that, imagery

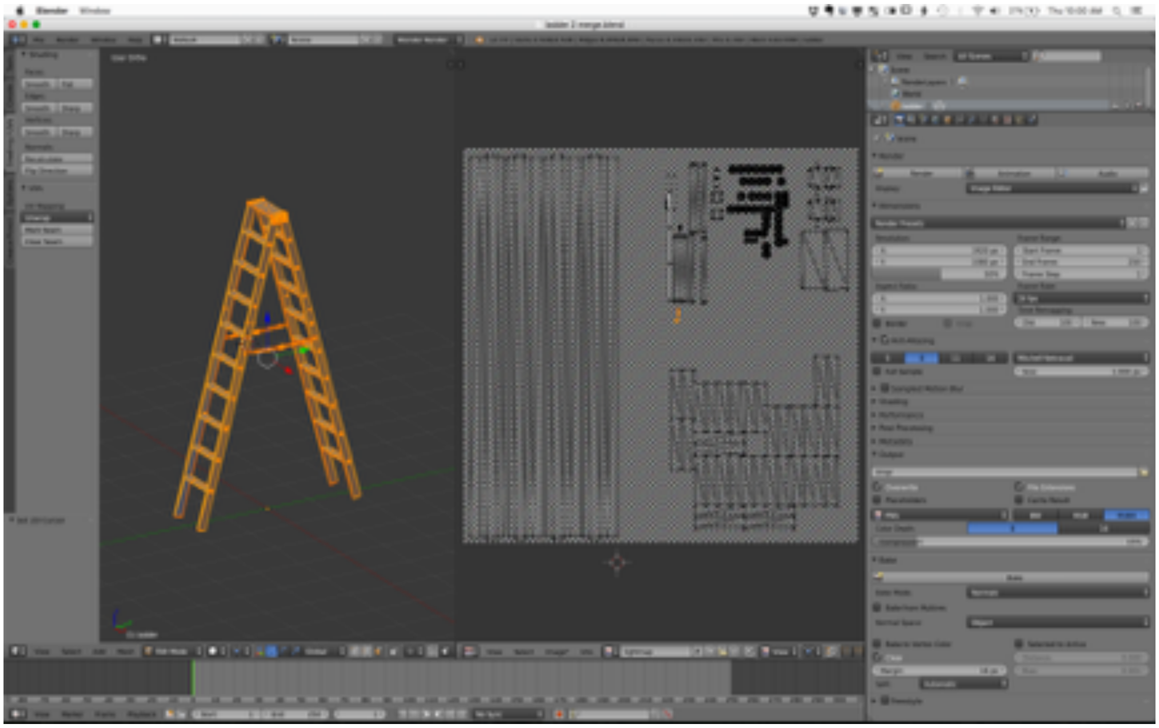

Figure 5: an unwrapped 3D object in Blender

can be applied to the 2D raster and it will map to the surfaces of the model. The main tool I used for this is a Photoshop plugin called 'Quixel'. Quixel is a professional-level piece of software used for texturing video game assets. It uses an intuitive color-coding system and impressive material libraries to generate textures for 3D objects. It makes texturing complex models a very quick and painless process (if the necessary material is in Quixel’s libraries). Otherwise, I must construct the textures myself and apply them to the model. During the course of this project I created textures for a variety of common theatrical materials such as muslin, scrim, duvetine, and stage decking.

Assembly

After all the ‘game’ assets (the term ‘asset’ refers to all the ‘pieces’ of a scene—such as 3D models, textures, audio files, etc.) are complete it comes time to build the scene inside of UE4. 3D objects are imported in a format called .FBX, texture files are all either .PNG or .JPG, and movie files are all .mp4 with an H.264 codec. After I’ve imported all the assets into the game engine, I began placing all of the elements in a process that is referred to as ‘greyboxing’, which involves placing all the different elements and some lighting but not applying textures. This process is similar to the scenic designer assembling a white paper model. It gives me and the team the chance to view all the objects and their spatial relationships to one another without worrying about color or texture. Once we’re satisfied with the placement of things, I begin building materials for each object in the scene. UE4 uses what is called a physically based renderer (PBR) for texture materials. PBR rendering creates photorealistic digital materials by mathematically controlling how light bounces off objects within the gaming engine. With this technology, UE4 creates a highly intuitive system for creating libraries of different materials that can be quickly applied to objects within the scene.

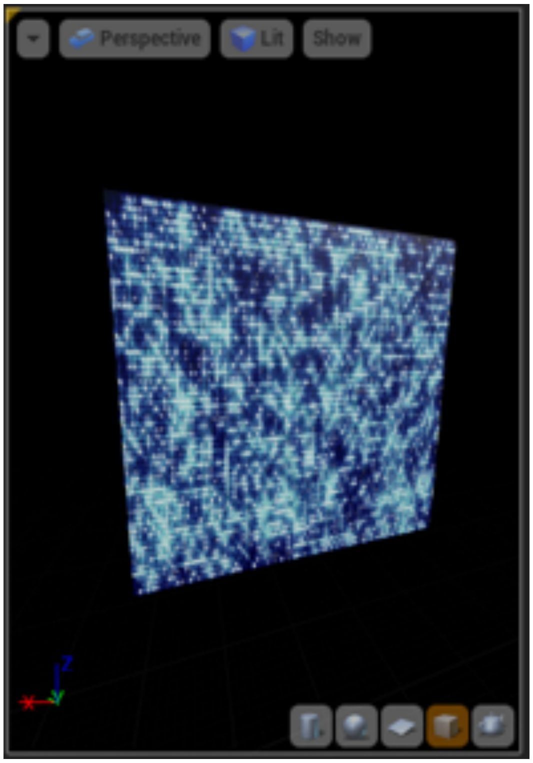

Figure 6: example of a material for an LED wall.

Lighting the Scene

Once the objects are placed and textured I began lighting the scene. In order for this tool to function properly I felt that the appearance of lights in the engine must be as close as possible to their real-world counterparts. An ETC Source 4 36° beam inside the engine should look as close as possible to a ETC Source 4 36° beam in reality. Fortunately, UE4 has a feature that allows the user to import IES beam profiles and apply them to lights. IES is the Illuminating Engineering Society, which is an organization of engineers and designers that develop standards and guides for the lighting industry. Every lamp produces a unique beam and IES developed a way to document these characteristics so they can be replicated digitally. This information is encapsulated into a file called an IES profile. Applying this file to a light object within UE4 will modify the appearance of the light beam to look like the light does in reality. As long as the manufacturer of a given light publishes their IES profiles I will be able to recreate them inside the engine.

In addition to creating lights I also created video lights and surfaces. Video lights are essentially lights that have their intensity filtered by a video file. With UE4's current set of tools this is the closest I can get to creating a projector tool within the engine. For video surfaces (such as LED walls, TVs, or monitors) I created materials that

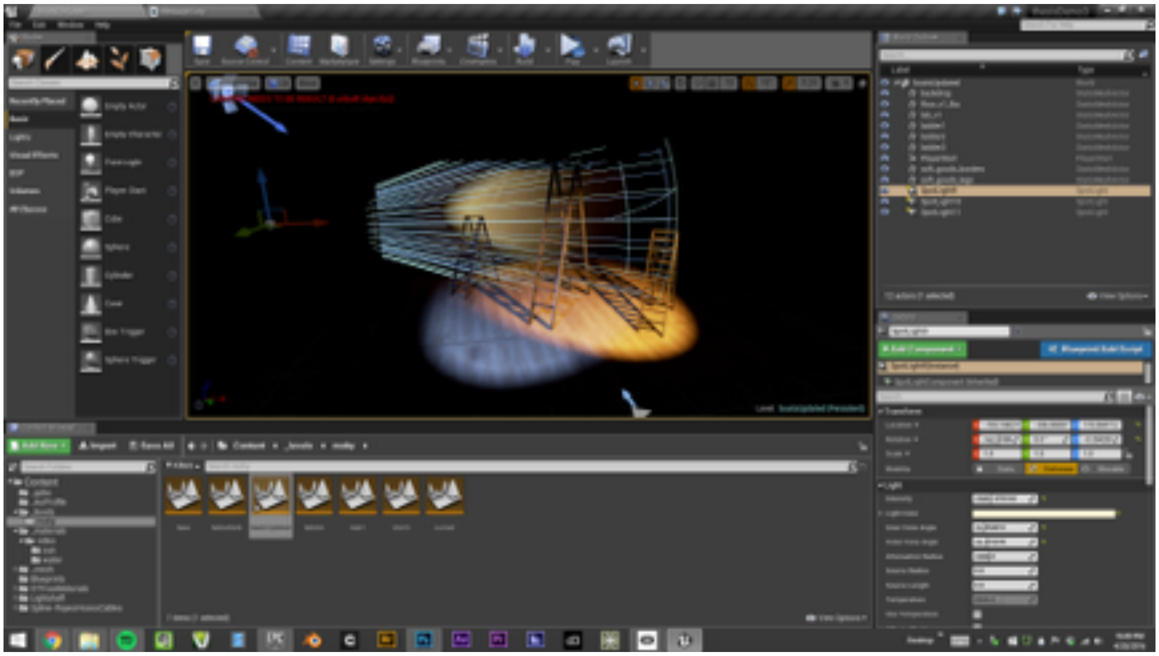

Figure 7: lighting the scene in UE4

included a small amount of emission. This gave the surfaces a ‘glowing’ appearance in the scene and also illuminated objects in close proximity to the surface. Brightness of the screen can be controlled via the material Blueprint assigned to the object. This gives the surfaces an uncanny resemblance to actual LED panels or monitors used in reality.

After all of these elements are assembled into the scene and the scene is lit it comes time to ‘build’ the lighting for the scene. This is a process in which the software pre-computes all of the lighting and shadows for the scene. This helps conserve computer resources and helps improve visual quality because the software no longer has to compute all the different lighting information strictly in real time. This process can take some time depending on the scale and level of detail in the scene and the available resources of the computer being used. This is an absolutely necessary part of the process but proved to be somewhat of a hindrance at times while trying to actively work in the scene during design meetings. This emphasized to me the importance of using high quality and fast computer components while working with UE4. Faster and higher quality components means less time waiting and an overall better graphical experience.

Optimization

This process also emphasized the importance of optimization of textures and objects in the scene. If a texture’s resolution is too high or a 3D object has too many vertices it is possible for the software to be unable to maintain a constant frame rate, resulting in a ‘stuttering’ effect. Virtual reality works at a much higher frame rate than typical video games, so any significant drop in frame rate will result in a poor product and potentially a very disoriented user. I discovered that while working with virtual reality in UE4 it was best to disable all default post-processing effects such as lens flare and light bloom as these effects can lower the overall frame rate. I also had to make decisions on how much detail to apply to objects based on their relative location within a scene. For instance, a texture and model for the ceiling or distant detail in the scene does not require nearly as much detail as an object that the user can walk right up to in the scene. At a distance, high resolution textures and high poly models simply don’t pay off and aren't necessary. These are all things to take into consideration while setting up scenes in virtual reality.

At this stage in development I felt I had gained sufficient skill and knowledge with the basic functions of the software and tools. I created several ‘demo scenes’ to explore different functions within the software in order to understand the considerations and potential limitations of the project. It was time to begin applying these ideas to a practical project with a real team of collaborators

Chapter 3: Virtual Reality Collaboration in Practice

A group of collaborators and I decided to devise a small theatrical production using virtual reality as the primary method of communication for the design process. My collaborators included Natalie Novacek (director), Christopher Kehoe (playwright), Teena Sauvola (scenic designer), Lirit Pendell (costume design), Matt Smith (lighting design), and Joe Heike (production manager). I decided to handle the projection and sound design for the project as I have a passion for both areas. Natalie and myself had been working on the idea for this production for over a year at this point and we decided that approaching it as part of our thesis studies would afford us the opportunity to examine our individual processes in much greater detail than we would normally have time for.

After much consideration and many discussions we decided to attempt creating a design-driven stage adaptation of the novel *Moby Dick* by Herman Melville. This aspect of the project is also the main performative element of Natalie Novacek's thesis project as well. She is interested in exploring design driven storytelling and how theatre makers can communicate story and emotion through visuals more than through text.[1] This began by having the designers involved with the script-writing process from the very beginning and having co-authorship over the text of the production. We felt this story lends itself to a process like this because the environment is essentially a character in the story. In this story the boat and the ocean embody danger, safety, beauty, and terror

[1] In her own words, "*Moby Dick/Book of Ishmael* is an subversion of the traditional new play development model. It calls into question who has authorship, how we prioritize visual storytelling, and what happens if we lead with design rather than text."

all at the same time. The characters are very much at the mercy of their environment. With this approach in mind I felt it was a golden opportunity to test the collaborative feasibility of virtual reality and video game engines as a design tool.

Our first step was to read the novel and discuss the themes and aspects of the story that spoke to us individually. At this point in the process, the team consisted of only me, Natalie Novacek, and Teena Sauvola (the rest would be recruited over time). We felt that it was imperative to start from a place of story rather than technology. After reading the novel, we each picked what we called our ‘greatest hits’, or the parts of the novel that we felt had both the most visual and storytelling potential. After picking these moments we dedicated and entire day to charting the story as we saw it so that we could begin visualizing the narrative of our adaptation. This chart was about 50 yards of paper where we created the entire timeline of the novel. We each placed markers at the moments that we each responded to the most, as well as placing research imagery in the timeline that we each felt related to a particular moment. This part of the process was essential to both developing the visual narrative and a cohesive vision for the world we wanted to create.

After these preliminary meetings Teena and I began designing the world inside of virtual reality. As previously mentioned, I designed my virtual reality workflow to be compatible with traditional design processes. As Teena began sketching ideas of shapes and forms that the stage could take, I translated them into 3d within computer software. Each iteration of these designs were then viewed inside of virtual reality and shared with the team until we found a design that we felt accurately portrayed our vision for the show and was technically feasible within our production limitations.

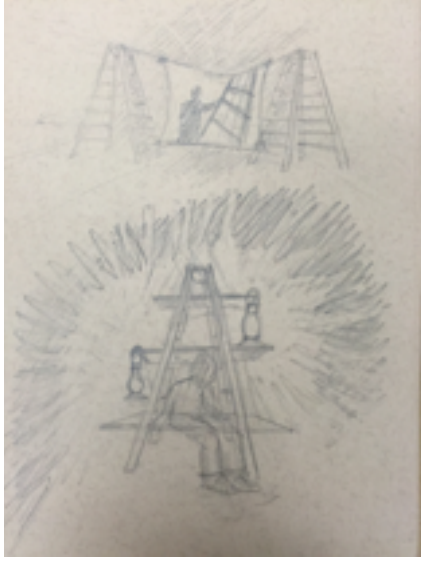

Figure 8: One of Teena's Sketches

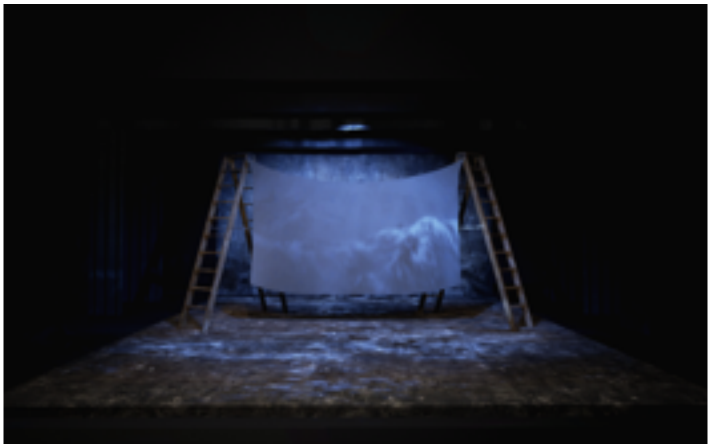

Figure 9: Teena's sketch translated into UE4

Once we had some ideas of the visual world we decided it was time to bring a playwright on board. At this point we had a string of moments that together could tell a story but we needed someone with more skill at creating texts to 'glue' the whole thing together. When first meeting the playwright (Christopher Kehoe) we discussed our approach to the process and showed images of what we had created inside of the software. At first this was a long distance collaboration so unfortunately it was impossible for Christopher to view the world inside of virtual reality as he did not own a virtual reality headset. Fortunately, UE4 is capable of outputting both movies and still images so the same visuals could be communicated, they just were not virtual reality. This was not ideal, and revealed a significant weakness of using virtual reality as a design tool— availability. At the time this paper is being written, the very first consumer model virtual reality headsets are shipping to customers and they come at a high premium ($600-$900 for the headset, not including a compatible PC capable of running the software). Given this, I simply cannot expect all of my collaborators to invest in a full virtual reality setup in order to participate in this process.

Figure 10: Director Natalie Novacek viewing a design in VR

This process repeated several times over the following months. Teena and I would produce visuals and Christopher would look at the visuals and revise the script based on how he responded to them. The result was a text that was heavily infused with visual language. There were many long and extraordinarily detailed sections of stage directions that were directly influenced by the visual world the design team created. The script had enough specificity to tell the story but was open-ended enough to allow for a lot of exploration and discovery in the rehearsal and tech process.

At this point we had a solid idea of the characteristics of the visual world, a script, a cast, and a space. We managed to secure a full month of time to load in, rehearse, and tech the scenes. For that full month all members of the team were regularly at rehearsals and we no longer required the use of VR for communication. Using virtual reality as a tool for the pre-production processes enabled us to enter the space with a solidified and cohesive vision and also helped solve many technical problems such as sightlines, projector positions, lighting positions, and scenic composition. There were very, very few surprises, which is something anyone working in the entertainment industry can appreciate. Every team member was able to enter the space with a very detailed understanding of when to expect in terms of the visual relationships between objects in the space. We were able to identify possible problems and discuss solutions well before ever starting rehearsals. For instance, in our initial visions of the design for the project we felt that creating a grand sense of scale was vital to helping the audience understand the sheer power of the environment in the story. Enormous, crashing waves and gigantic leviathans were all part of the initial visual language. When we could not secure a space that could accommodate for this scale, we were able to re-envision our story in a much smaller space with these limitations in mind. We chose to try and theatricalize elements of the story and create a sense of scale on the floor of the space instead. This is a problem we may not have immediately identified if we had not had the ability to virtually explore the space ahead of time.

In general, the response to the use of virtual reality was enormously positive from all members of the creative team. Outside of the aforementioned disadvantages like availability, everyone seemed amazed at the collaborative possibilities the software offers. One professional director I showed the project to noted that she could block her entire show ahead of time using this tool. Not only would she be able to view the stage as an audience member, but she could also place herself in the position of the actor and get a sense of what they will see in a particular moment. Another collaborator noted the following:

> “The impact on storytelling is monumental, because directors have the opportunity to consider how the set, lighting, media, etc. can be used in multiple ways to create the world of the play, rather than just serving as a backdrop. Lastly, the technology will hopefully avoid those moments during tech rehearsals where you sometimes realize "it just won't work".

I found this piece of feedback particularly inspiring as I think it falls directly in line with what I wanted the technology to be able to achieve. It is a tool that offers the possibility of both solving problems and opening possibilities in the early and crucial stages of a design process. There were concerns as well, one collaborator noted that he “…hopes that the technology doesn't lock us into one way of solving a problem, or lock us into one way of thinking something has to be done in the space.” I find this to be a very valid concern. A lot of things can change once bodies are in an actual physical space. The advantages of solving problems beforehand in VR should not overshadow or inhibit discovery made in the physical space.

Chapter 4: Conclusion

I also displayed my studies at the University of Texas at Austin Center for Arts and Technology launch event in February of 2016. This afforded me the opportunity to show the work to a diversity of individuals both in and out of the entertainment industry. Everyone I spoke to seemed very enthusiastic about the possibilities of this technology. One individual I spoke with was part of a major real estate company and was very enthusiastic about the possibilities for his industry as it would give potential buyers the opportunity to view a piece of real estate virtually before committing to a purchase. They could even design their own interiors to get a better sense of how the space could be used. This and other conversations revealed to me the very broad range of possibilities that come along with this technology, both in and out of the entertainment industry.

Other professionals and companies have been begun embracing the creative possibilities of this technology as well, with companies such as Moment Factory and Industrial Light and Magic announcing the use of VR to pre-visualize sets and experiences for their clients. A designer and art director in the theme park industry, Don Carson, referred to virtual reality as a ‘quiet design revolution’. He notes how this technology has a great deal of potential to fundamentally change how design teams view their ideas, but no one seems to be talking about it. The focus of the virtual reality market is highly targeted on gaming and media consumption at the moment, and it is my hope that projects such as this one can illuminate the utility of virtual reality as a design and collaboration tool.

I had previously mentioned how availability of VR is an issue at the moment, but there may be a possible solution in the short term. Google released a VR device called ‘Google Cardboard’. It is literally a cardboard VR viewfinder with rudimentary optics and slot where the user can attach their smartphone. There are apps that can be downloaded that will display VR content through the phone. Cardboard can be purchased for as little as $10 and considering the current popularity of smartphones it is very likely that most collaborators would already have a compatible phone. Due to the limited computing resources of smart phones, it means that I would have to create an entirely separate project with lower quality models and textures in order to work this way, so I did not have the chance to utilize this method for this particular project. It does not have the

Figure 11: Google Cardboard

same amount of interactive potential, but it is a possible stopgap until VR hardware becomes more affordable and available in the marketplace.

Personally I find this technology to be incredibly inspiring to work with. There is an interactive and almost tactile quality to viewing designs inside of virtual reality that I long for while working with two-dimensional media. It allows me to immerse myself into a space and fully grasp the relationships between forms and gain and understanding of scale within the design. There are many drawbacks, such as the cost of entry, availability, and a steep learning curve, but I believe that as the technology continues to develop and expand these shortcomings will become negligible. I look forward to the continuing to study and leverage the creative possibilities virtual reality has to offer.

References

Blender Manual Contents. (n.d.). Retrieved from https://www.blender.org/manual/

Carson, D. (2015, February 4). Don Carson Blog: Theme Park 101 - Oculus Rift - A Quiet Design Revolution. Retrieved from http://themedenvironments.blogspot.com/2015/02/theme-park-101-oculus-rift-quiet-design.html

Game Engine Technology by Unreal. (n.d.). Retrieved from https://www.unrealengine.-com/blog

Robinson, M. (2016, April 13.) Tech Insider: Here's how the makers of 'Rogue One' used VR to create the film. Retrieved from http://www.techinsider.io/ilm-virtual-reality-vr-rogue-one-2016-4

Unreal Engine 4 Documentation. (n.d.). Retrieved from https://docs.unrealengine.com/latest/INT/