2015

The Significance of Previsualization Software in Academia

Joshua E. Taylor

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Part of the Other Theatre and Performance Studies Commons

Recommended Citation
https://researchrepository.wvu.edu/etd/6779

This Thesis is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Thesis has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.
The Significance of Previsualization Software in Academia

Joshua E. Taylor

Thesis submitted
to the College of Creative Arts
at West Virginia University

in partial fulfillment of the requirements for the degree of

Master of Fine Arts
in
Theatre

Committee Members:
Steven Neuenschwander (Chair), Clinical Assistant Professor of Technical Direction and Production Manager
Alan McEwen, Clinical Assistant Professor of Lighting and Sound Design
Joshua Williamson, Director and Associate Professor of Theatre

School of Theatre & Dance

Morgantown, WV
2015

Keywords: Technical Theatre, Lighting, Visualizer, Previsualization, Stage Lighting, Entertainment Lighting, Lighting Design, 3D Rendering

Copyright 2015 Joshua Taylor
Abstract

The Significance of Previsualization Software in Academia

Joshua Taylor

The following body of work serves as a guide to setting up and effectively using entertainment lighting previsualization software as a teaching tool in higher education. The ultimate goal is to bring this lighting console training to students at a greatly reduced cost. Topics covered include: the pros and cons of a variety of visualizers, the equipment necessary, additional software that works in conjunction with the visualizers, and an outline of my Advanced Programming special topics course taught in conjunction with this study.
# TABLE OF CONTENTS

ABSTRACT ........................................................................................................................................... I

TABLE OF FIGURES ............................................................................................................................... III

ACKNOWLEDGMENTS ............................................................................................................................... V

CHAPTER 1.  INTRODUCTION ............................................................................................................... 1

CHAPTER 2.  VISUALIZERS .................................................................................................................... 4

  WYSIWYG ........................................................................................................................................... 5
  CAPTURE SWEDEN ............................................................................................................................... 7
  LIGHTCONVERSE .............................................................................................................................. 9
  GRANDMA 3D ................................................................................................................................... 11

CHAPTER 3.  ESP VISION ........................................................................................................................ 13

  OVERVIEW ......................................................................................................................................... 16
    The Work Space ................................................................................................................................. 16
    Toolbars ........................................................................................................................................... 17
    Layers and Properties ........................................................................................................................ 19
  VECTORWORKS ............................................................................................................................... 21

CHAPTER 4.  CONSOLE HOOKUP ........................................................................................................ 26

  ETC NOMAD ...................................................................................................................................... 26
  HOG ................................................................................................................................................... 28
  GRANDMA ......................................................................................................................................... 29
  CONNECTION CHECKLISTS ............................................................................................................... 31

CHAPTER 5.  MY CLASS .......................................................................................................................... 34
CHAPTER 6. THE CLASSROOM ............................................................................................................. 37
CHAPTER 7. COMPARISONS .................................................................................................................. 42
SHOW PREVISUALIZATION .................................................................................................................. 43
ACCURACY COMPARISON .................................................................................................................... 46
KEYBOARD/ FACE PANEL FORMATTING .......................................................................................... 48
ETC Ion .................................................................................................................................................. 48
High End Hog4 ..................................................................................................................................... 49
GrandMA2 ........................................................................................................................................... 50
CHAPTER 8. FINAL THOUGHTS ............................................................................................................ 51
APPENDIX I: RISER DIAGRAM ............................................................................................................ 53
APPENDIX II: GLOSSARY OF TERMS ............................................................................................... 54
APPENDIX III: SYSTEM REQUIREMENTS ......................................................................................... 56
WYSIWYG ............................................................................................................................................... 56
CAPTURE SWEDEN ............................................................................................................................... 56
LIGHTCONVERSE ............................................................................................................................... 57
GRANDMA 3D ....................................................................................................................................... 57
APPENDIX IV: STUDENT WORK .......................................................................................................... 58
APPENDIX V ........................................................................................................................................... 60
BIBLIOGRAPHY ..................................................................................................................................... 63
WORKS CITED ....................................................................................................................................... 65
# Table of Figures

- **Figure 2.1**: WYSIWYG work screen (Cast) ................................................................. 5
- **Figure 2.2**: Capture Polar work screen ([image](0_image_large_538.gif)) ............ 7
- **Figure 2.3**: LightConverse work screen ([lc1.jpg](Lc1.jpg) – Scaled (63%)) .......... 9
- **Figure 2.4**: GrandMA 3d work screen ([grandMA 3D](grandMA 3D | LTT)) ............... 11
- **Figure 3.1**: Main screen set up with four viewports .................................................. 16
- **Figure 3.2**: Right click drop down for view options .................................................... 16
- **Figure 3.3**: Tools toolbar, Camera toolbar, Modify toolbar ........................................ 17
- **Figure 3.4**: AddInst selection screen ........................................................................... 18
- **Figure 3.5**: Scene Graph and Properties ...................................................................... 19
- **Figure 3.6**: Controlable properties of a lighting fixture ................................................ 20
- **Figure 3.7**: Workspace Editor within Vectorworks ....................................................... 22
- **Figure 3.8**: Vision drop down menu ............................................................................ 23
- **Figure 3.9**: ESP Vision light editor window ................................................................. 24
- **Figure 4.1**: Internet Protocol Version 4 window ........................................................... 27
- **Figure 4.2**: Nomad Start-up Screen ............................................................................ 27
- **Figure 4.3**: ETC EOS Family Network window ............................................................ 28
- **Figure 4.4**: DMX Provider selection menu ................................................................. 28
- **Figure 4.5**: Hog net connection window ..................................................................... 29
- **Figure 4.6**: Main screen ............................................................................................. 30
- **Figure 4.7**: GrandMA network configuration window ............................................... 30
- **Figure 4.8**: GrandMA to ESP Vision plug-in window .................................................. 31
- **Figure 6.1**: Alienware X51-R2 (Delaney) ................................................................. 38
- **Figure 6.2**: Cherry SPOS Rows and Columns Keyboard (CHERRY: Innovation at Your Fingertips- Spos Rows & Columns Keyboard) .................................................. 39
Figure 6.3: Four Monitor Stand (STAND-V004F.jpg (JPEG Image, 435 X 600 pixels)) ........................................ 40

Figure 6.4: Full set up at West Virginia University including: Alienware X51-R2, four monitors, monitor tower, Cherry Rows and Columns programmable keyboard, and VGA splitter. ................................................................. 41

Figure 7.1: ESP Vision Marketing Photo (ESP Vision: Vision Pro) ........................................................................ 43

Figure 7.2: The Glass Menagerie “Laura Alone” rendering .................................................................................. 44

Figure 7.3: The Glass Menagerie “Laura Alone” .................................................................................................. 44

Figure 7.4: The Glass Menagerie “Toms and Amanda” rendering ....................................................................... 45

Figure 7.5: The Glass Menagerie “Toms Lament” ............................................................................................... 45

Figure 7.6: Notice the blue square in the space, used as a point of reference. (Rendering) ............................... 46

Figure 7.7: Notice the bottom edge of the taped out square in red. ................................................................. 46

Figure 7.8: Notice the color wheel split between colors. (Rendering) .............................................................. 47

Figure 7.9: Notice the same split at the bottom of the beam. ............................................................................. 47

Figure 7.10: Ion keyboard layout ...................................................................................................................... 48

Figure 7.11: Ion face panel .................................................................................................................................. 48

Figure 7.12: Hog4 keyboard layout ...................................................................................................................... 49

Figure 7.13: Hog4 virtual face panel ................................................................................................................... 49

Figure 7.14: GrandMA2 keyboard layout .......................................................................................................... 50

Figure 7.15: GrandMA2 Face panel ................................................................................................................... 50

Figure IV.1: Daniel Delbusto’s work from The Who’s Tommy ................................................................. 58

Figure IV.2: Rachel Lake’s work from a concert of Florence + The Machine songs ..................................... 59

Figure IV.3: Savannah Yost’s work from American Idiot ................................................................................. 59
Acknowledgments

I wish to thank my mom for all of her hard work, patience, and support.

I would also like to thank Chad Nulik for teaching me that anything is possible with hard work and a little creative thinking.

I would like to thank Ron Fowlkes, Nancy Pontious, and Jim Bartruff at Emporia State University for the inspiration and support.

I would like to thank Alan McEwen for his continuous support and guidance while here at West Virginia University.

I would like to thank Ross Rundell, Tim Thistleton, and Ryan Delbridge for their companionship and encouragement. They have made the past few years so much easier.

Lastly, I wish to thank Brianne Taylor, without her I would not be here. Without her I would not be who I am today. Most of all, without her I would not be happy.
Chapter 1. Introduction

Previsualization software is a computer program that is used to set up and manipulate a 3D space, much like a modern CAD program. CAD programs are primarily used in theatre for creating ground plans, light plots, and working drawings. However in recent years, manufacturers of these programs have improved the quality of their 3D rendering capabilities. With the advent of moving fixtures, a static image does not always convey the entire thought. Rather, the design may be better demonstrated with a video to show how the movement of the light affects the atmosphere within the space.

One of the largest challenges for the lighting designer is time. Lighting is typically given a short amount of time once in the theatre to set up and prepare cues for the quickly approaching technical rehearsal. The short time frame is because scenery needs to get set up and lighting spends most of the time in the dark either focusing lights or creating cues. There is only so much pre-tech work that can be done by the designer outside of the venue including: patch, creating pallets, presets, groups, and creating effects.

However, with the ubiquity of computers, the lighting designer’s job has become less tedious than before. Not only can we store cues into a lighting console, but we can also make some of those cues at home on an offline editor. Offline editors have also dramatically helped the lighting programmer. At any time, on-the-go or at home, the programmer can whip out his or her laptop and work on a show, whether getting a current show ready or by prepping for the next show. There is no need to be physically in the space to preprogram a show and there is no need to go down to the local rental house and
use their console. The programmer can do a significant amount of work at home with the offline software.

For some designers, this new option to work outside the venue is great. Their minds work well with a graphical representation of the lighting design. For others, programming blind can be a very difficult task, creating cues without visual representation of the stage and lights leaves the programmer guessing what the scene may look like. This is where the visualizer comes in--a computer application which displays a virtual stage and lighting equipment. This tool brings the stage and lighting equipment to the computer screen and increases the effectiveness of the offline editor greatly. The programmer and lighting designer can now be anywhere with a computer, type in data and see results on the screen. With real-time representation of work being done, the design possibilities broaden. While programming blind a designer could set color and beam presets; however, now we can even set focus presets using a rendering of the theatre to reference. Before, designers had to rely on instinct and imagination to set up cues. Now they can get a glimpse of the show they are designing.

For the professional, this tool can be incredibly effective in today’s fast-paced world. The ability to lay down such a large foundation of pre-production work makes for a much smoother technical rehearsal or at the very least cuts down on the programming time needed in the space. Instead of getting a show finished just in time, designers can depend on a foundation of cues and presets to build on, ultimately resulting in a better final product. Visualizers brought the theatre to the lighting designers and helped them show clients and directors what certain scenes may look like. When accurately rendered,
the visualizer can help target possible issues that could surprise the designer once in the space. But what if this tool could be used for more than pre-production work?

Lighting consoles in the entertainment industry are costly. For most, training on these consoles comes from years on the road; and for others it is costly training at tradeshows or at the company. It is no secret that theatre training programs have limited funds, particularly for lighting. This is where higher education can take advantage of previsualization software--by bringing these consoles and other equipment into the classroom to be studied and trained on. Instead of a $50,000 console, classes can use the offline editor and connect it to previsualization software. The cost is quickly diminished to a couple thousand dollars for a new computer and the visualization software. This allows the student to experience the console and understand how it works on a fundamental level. The syntax, tips, and tricks can be discovered and honed in a classroom setting. Students and professors are not limited to their own inventory but are instead allotted an expansive inventory of both old and new equipment to use on their own console or offline editor.

There are factors to consider before plunging into a project of this magnitude, such as time, space, money, and current applications used in the theatre program. This report offers a list of available previsualization software being used in the entertainment industry including some of their highlights. The following report explores the layout, function, networking procedures, and performance for ESP Vision, the previsualisation software in use at West Virginia University and an outline of the Advance Programming special topics course being taught at West Virginia University School of Theatre & Dance.
Chapter 2. Visualizers

There are a few different visualizers on the market and all offer a variety of different functions. Each one is unique in its own way but all are equally effective as a design and teaching tool. The cost varies across all companies, but the adage stands true: you get what you pay for. That being said, some companies offer free or student versions of their software with limited capabilities.

In a constantly growing and changing industry, visualizer companies are constantly working to keep up with fixture profiles (DMX channel layouts) of new lights for their software, which means there can be frequent, even daily updates that should be installed. After the initial cost of purchasing the software it is likely you will be asked to pay a yearly subscription fee. If your subscription lapses then you may be cut off from the daily updates and/or product support.

Visualizers take 3D rendering to a new level. Not only can you render lighting in the space but you can also send live data from the computer or lighting console and see lighting cues in real time. This means you can show a director or a client many scenes in motion rather than a static image. While all visualizers are different in appearance they all function essentially the same. The venue is rendered in OpenGL (rendering engine) or a similar format, depending on the platform. Stage elements, like people, scenery, props and curtains, are all added to the space as block units. Their geometric detail is relatively low, more boxy and rigid than realistic. This is important to keep the programs from processing too much unnecessary data. A good way to understand this concept is to look at a square versus a circle. Squares only have four points to process whereas a computer sees a circle as an infinite number of points to process.
When theatre lighting practitioners think of previsualisation, the common thought is WYSIWYG (What You See Is What You Get): a company that has been around since the 1990’s and has developed quite a reputation. When starting this project I was surprised at the number of companies that offered visualizers.

WYSIWYG

WYSIWYG is perhaps one of the most well-known previsualisation products in the entertainment industry. When the company started in the mid-nineties it was to satisfy the industry’s two very important needs: a program for previsualisation and CAD software dedicated to entertainment lighting design. For close to half a decade WYSIWYG dominated the market, but over time it developed a bad reputation due to poor customer service. Today, they still have arguably the most recognizable name in the

![Figure 2.1: WYSIWYG work screen (Cast)](image)
industry and have built an incredible product that certainly meets today’s standards.

WYSIWYG is on its 34th release and in some ways still leads the curve in entertainment previsualization. Brand new with this release is the use of lasers, which can be connected to a laser control system or set to a few different default looks to be included in the final renderings. Another rather large step forward for WYSIWYG is the ability to import Sketchup 2015 files. This is crucial because the user no longer needs to do all rendering and CAD work with the proprietary CAD of WYSIWYG. The CAD portion of the software is perhaps the weakest facet of the program. In the mid-nineties when the only other alternative was MiniCAD, this new CAD program was a great improvement and made life a little easier. Now, with Vectorworks and Lightwright setting industry standards, this ability to do paperwork seems obsolete. However, if the user has no experience with other programs and is looking for an all-inclusive package, WYSIWYG is a viable option.

As far as WYSIWYG’s performance, it is certainly on par with other visualizers and in some cases better. The quality of light seems natural and real; shadows and texture render perfectly. Most objects in the space appear blocky and fake but definitely get the point across. With the release of WYSIWYG R34 there are some new ambient lighting settings that are really intuitive and make a more realistic atmosphere. Though that may sound silly or unnecessary, it is the little things that make the difference when showing a director a video on a screen and conveying the sense of an intimate and artistic design.

WYSIWYG is a great option for those looking for a program to visualize what scenes may look like, because the program produces an incredibly high-quality image
with vast control of the atmospheric elements of the scene. When the rendering is set
side by side to a real picture of the same scene the likeness is truly remarkable.

Cast’s WYSIWYG is a great program that has grown tremendously over time, it
makes great advances with each release, and will undoubtedly continue to do so in the
future. The high quality rendering comes at a high price, though. To design, render, and
have a live network connection with a console it will currently cost $5,749 for the
professional and $529 for the student both of which are Windows only. For more
information and purchasing visit http://cast-soft.com.

Capture Sweden

Capture is a lighting previsualization company based out of Härryda, Sweden.
Capture also started in the early nineties for the same reason as WYSIWYG: to make
computer software to aid entertainment lighting designers.

Figure 2.2: Capture Polar work screen (0_image_large_538.gif (GIF Image, 1100 X 725 pixels))
I find the demo version incredibly useful when teaching pixel mapping because it is quick and easy to place and arrange the fixtures within the space, sequentially patch, and connect to an external lighting console. The demo version comes with a substantial library and most of the bells and whistles. The major disadvantage is that show files created cannot be saved and the software closes after thirty minutes.

There are a few different purchasing options to choose from but you certainly get what you pay for. The student version is by far the cheapest package costing the user nothing. However, the fixture library is quite limited which doesn’t allow the user to see and use new equipment. The software can still work live with a console and is not locked on a stock show file, so the user is free to design and program within the given limitations.

The solo version is the first of the pay-to-use packages and includes a full library, lasers, and video playback. Considering the cost is only $395 dollars this is a strong contender for a lighting visualizer in the academic setting. The drawback to this package is that it only works with a single universe of DMX 512. For a thousand dollars more, the next package includes paperwork and unlimited universes. Considering you may have access to Vectorworks and Lightwright, this option may not be worth the cost. The final package costs $2,895 and includes a great deal of additions including the ability to import DXF and DWG files, moving scenery, stream video from media servers, rendering movies, and water jets.

Capture is incredibly easy to pick up and use on the fly and fits within the company’s mantra: “If a user doesn’t instantly understand how to use a feature, it's not
good enough.”\textsuperscript{1} As a free student version, Capture is a great tool for the ambitious student to download and practice using at home or in class. However the solo version is a cost effective way to bring thousands of moving lights and LED fixtures into the classroom to be used. Capture is both Windows and Mac compatible, and includes paperwork, plotting functions, and could be the right fit for someone looking for an all-inclusive package. For more information and purchasing visit http://www.capturesweden.com/Buy/Licensing.aspx.

**LightConverse**

LightConverse is a program that got its start in the late nineties in Ukraine and debuted in North America in 2007 at Live Design International. This software is PC only and renders via a DirectX interface. There are also 2D plotting possibilities within the

\textsuperscript{1} Berg
program, but this falls short of today’s standards. Plugins are available however for importing from AutoCAD, Vectorworks, and Google SketchUp. One thing that sets LightConverse apart from the other visualizers is its “Fixture Construction” capabilities. This useful tool allows users to get their hands on a brand new fixture before the company is able to release its profile. For more information and purchasing visit http://lightconverse.net/.

The starting price for a version of LightConverse that can network with external consoles is roughly $800 before shipping, making it a possibility for a university or a young professional, but a little expensive for the average student. Also available for purchase is a LightConverse-Server, a large computer with a special version of the software to enhance the programs video input and output capabilities. LightConverse has a realistic image quality, much like WYSIWYG, when finally rendered and even when working within the space it performs with the efficiency of Capture or ESP Vision. One minor drawback is the user interface; it is a little busy (Figure 2.3). Other than that, the program performs as well as other visualizers and should definitely be considered when shopping for visualization software.
GrandMA 3D

GrandMA3D is a great visualizer with the only downside being that it is exclusively usable by grandMA consoles. One of the best advantages of MA3D is that when patching a show on the console the lights populate the MA3D application as well.

This is especially useful considering you can give fixtures X, Y, and Z coordinates in the patch that will be reflected in MA3D. The program includes an extensive media database full of 3D objects ranging from letters, people, instruments, and plenty of other stage elements. There are also a handful of light rendering settings that can be adjusted to make for a more aesthetically realistic final view, whether it is a simple screenshot or to record a video. For the end user focused on learning or improving skills on grandMA systems specifically, this is the best option considering it is free and easy to use. For

These visualizers are all different in their own way but are equally effective tools for a lighting designer and especially for a student. Some come at a steep price, but considering what the programs are doing the cost is well worth it. West Virginia University’s School of Theatre & Dance decided to use ESP Vision for its visualization needs, and the reasons are detailed in the next chapter. I have only touched on their individual function, mainly because some of their other functions are not all that useful in educational theatre. All of the pay-to-use visualizers include the ability to connect to automated scenery controls and media servers; and render pyro, fireworks, and fountains. For the most part, all of the visualizers also connect to external consoles for control using ethernet. Ultimately, it is important to use this information as a jumping-off point and to conduct your own research to see which visualizer would best suit your needs.
Chapter 3. ESP Vision

In the fall of 2013, I began to research visualizers that could be used as teaching tools for a special topics class to be taught in the spring of 2015. This search turned out to be a challenging task. Step one of the research was to assemble a list of available visualizers on the market. This part was not all that difficult considering a quick Google search of “lighting visualization software” produces WYSIWYG, Capture, LightConverse, and ESP Vision. With a list of visualizers to reference I could take a closer look at each individually.

Certain visualizers were dismissed solely due to the price of the product such as WYSIWYG. Others were originally considered then later dismissed based off of need and personal taste such as Capture Sweden and LightConverse. Major deciding factors were how difficult was the program to learn and use, work space aesthetic, image quality, and live performance. The school already had Vectorworks and Lightwright, so paperwork was not a necessary feature. The end goal of my research was to have a visual representation of theatrical lighting for the student to reference while learning and refining programming skills on a theatrical light console. There was no desire for high end, costly photorealistic renderings because the final goal was to improve programming, not necessarily to previsualize shows. When all was tested, calculated, and considered the final choice was ESP Vision for several reasons.

ESP Vision is a company based out of Las Vegas that started in 2000. They were originally a WYSIWYG Certified Service Provider. Members of the ESP Vision became frustrated with the product WYSIWYG was marketing and wanted to make something better. At the time, the founders discovered using DirectX for their rendering platform
made for a really nice product, as far as image quality and performance, and by 2004 they began to market their own visualizer. By 2008 the company noticed a big shift in the industry toward Mac products, so they switched their rendering engine to OpenGL to create cross-platform visualizer. Becoming a cross-platform program was helpful because it meant only one program to update, which meant less work on their end.

When all was said and done ESP Vision was the final choice due to ease of use and superior customer support. ESP Vision was the best at customer support I have come across while researching the visualizers. Other companies responded to emails, the earliest being twenty minutes and latest a few days later, so it is not to say they weren’t responsive, it just took some longer than others. This was a problem considering how difficult the final goal was to achieve: having three different offline editors sending live data to the visualizer, and switching between the three quickly and easily. ESP Vision beat out the competitors due to their small company size and their webpage live chat feature for technical support.

WYSIWYG also has a live chat function on their webpage, but the questions that could be answered are more for sales and not for technical support. Having a software expert just a few clicks away made the whole experience much easier. This was not only helpful during set up of this application but will doubtlessly prove helpful in the future. The chat function corresponds to normal business hours and the staff is helpful, personable, and kind. Oftentimes I chatted with the same person I had talked to before so it was easy to quickly pick up where we had left off. Representatives were happy to help with all matters related to using the visualizer, from networking issues to executing some basic functions.
Ease of use was another important factor when making the final decision. After downloading the demo I found it easy to just pick up and go. ESP Vision’s ultimate function is to render lighting data live using an external controller. This means the user interface is incredibly simple. There are only a handful of tools in the program and no CAD tools to clutter the space. The demo is a premade show, works as is, and has limited functionality, so I didn’t find out until we purchased the software that as a stand-alone program it can be clunky to use. This is because it is almost too easy to use; therefore, it does not accept most efficient computer functions like copy, paste, or cut; instead, it is point-and-click.

ESP Vision also offers almost daily updates to the software. It is recommended that you run the updater before each use to make sure all operating files are current. This feature says a lot about the quality of product the company is putting out. Updates include bug fixes, fixture updates including the addition of new fixtures or improvement of the quality of older fixtures as well as any additions and improvements to the software.

ESP Vision comes at three different costs: a “pro” option for $750, a “teach” option for $495 for schools and universities, and a “learn” option for $250 for students. All three options allow use of a single universe of DMX512, but more can be purchased for $750 a universe. As mentioned before, there is a yearly subscription fee of $150 which allows the update function to keep working as well as customer support. The original purchase includes the first subscription fee. To learn more about purchasing visit http://shop.espvision.com/Vision-Products_c2.htm.
Overview

ESP Vision’s sole purpose is to be used in conjunction with some form of external control, whether that be an offline editor or an actual lighting console. This is made clear considering it is impossible to preset lights at intensities or control moving lights within the visualizer. The lights will highlight when selected so they can be focused, but once you move on to the next light the previous snaps off. This is important to note if you are looking for standalone rendering software to create photos to convey ideas to clients, this may not be the right visualizer for you.

The Work Space

Upon launching the software the user is greeted with four different black screens. These are different views of an empty space called “viewports” (Figure 3.1). At the top left of each viewport there is identification of the view it is in. By right-clicking within that panel a drop down menu appears giving different view angles and options (Figure 3.2). Surrounding the views are a series of toolbars at the top and two menu sections to the left of the screen. The placement of these toolbars and menus are set by default and can be resized and moved by the user to best suit their needs. Movement and sizing can be adjusted the same as most
other computer applications, by locating adjustment points at the corners then clicking and dragging. The menu bar along the top of the screen has the standard File, Edit, and Help options but also includes Window, Render, and DMX. By clicking Window, one can open and close toolbars or menus. There is also a Toggle Number of Viewports option, which toggles between four view ports and one.

**Toolbars**

![Figure 3.3: Tools toolbar, Camera toolbar, Modify toolbar](image)

Along the top of the screen there is a series of toolbars, each with tool sets that can be added or removed depending on the user’s need. By default these are aligned across the top of the screen. The tool bars are separated into Tools on the left, Camera in the center, and Modify on the right. Each toolbar can be moved and arranged simply by clicking and dragging, and there are parts of the screen where the toolbars will snap or they can float anywhere on the screen.

The arrow icon is the selection tool followed by Sphere, Cube, Cylinder, Line, and Point tools. These shape tools are incredibly clunky and nearly useless considering what is to be accomplished and can be overlooked. This is because most of the building of the environment can be done in Vectorworks. Following the shape tools are Undo, Redo, Cut, Copy, and Paste. These function the same as any other application with exception of responding to traditional keyboard shortcuts.
Next, AddInst is a graphic of a light used to add fixtures to the space. When you click the tool, users are greeted by a pop-up box (Figure 3:4) that has three drop down menus. These menus make it easier to find the fixtures needed. Once found and selected, you can assign patch information and set a trim height if you so desire. By pressing OK the pop up disappears and the fixture is now ready to be placed in the space by simply clicking on the location desired.

After the AddInst tool there is an AddObj tool and works similarly to the AddInst tool. Click the tool and an Add Object Tool pop-up appears and looks the exact same as the previous pop-up. The drop-down menus separate all of the stock objects into Furniture, Miscellaneous, People, Pre-Built Rooms, Stage and Stage Elements, Truss, and Walls. From here users can set a trim height and a rotation to the object if necessary. Same as before, you can exit the window by pressing “OK” and place the object by clicking on the desired location.

Following the Tools toolbar there is the Camera toolbar with three different tools: Zoom, Pan, and Rotate. These tools function like one would assume by click and dragging or scrolling with the mouse wheel. All of these actions can be achieved by a mouse with a wheel where wheel scrolling is Zoom, wheel click and move is Pan, and Alt + Wheel click and move is Rotate.
The next set are the Modify tools which work about as clunky as the shape tools mentioned above. First of the three is the Move tool, which when activated, allows the movement of an object by clicking and dragging that object in any direction. Following this is the Scale tool which is used the exact same way, only making the object larger or smaller in whatever direction desired. Last of the three is the Rotate tool which is used in the same way as the previous two, but Rotates the object in all three planes as desired.

**Layers and Properties**

By default there are two menus to the left of the screen but like before they can snap to a new location or move wherever best suits the user’s needs. Under the Window menu at the top of the screen there is a Layer option to toggle the Scene Graph window on or off. Scene Graph (Figure 3.5) is where all of the objects and lights are stored for viewing and selection. This may be the most difficult and
possibly worst feature of ESP Vision because the navigation and properties of this menu are hard to get used to. ROOT is essentially the “space” users work in. By clicking the arrow it opens a sub tree called “New Layer” which also opens a sub tree with all of the objects within the space. All of these objects have sub trees with different bits of information, such as position and rotation settings along with texture if any. Selecting a menu or object reveals any properties it may have in the Properties menu.

Properties for the ROOT menu (Figure 3.5) are global and will affect how the space looks as a whole. The key properties here are Ambient (adjusting how light or dark the environment is), Volumetric Settings (quality and amount of light from fixtures), and Exposure. Selecting a fixture from a sub tree will have different properties as the Root menu. Figure 3.6 shows the properties for a Mac 2K Wash fixture. This is where users can view and make patch changes as well as adjust the intensity (candela) for that particular fixture. The color wheel and gobo wheel options are most useful when working with conventional fixtures because this is where users can add color and texture to the lights. When an object is selected, there are far fewer properties to adjust. Along
with position and rotation properties, there is a texture property that can be changed to affect the lights physical texture.

The **Window** menu has a couple other important tools to help set up and run shows within the application, first of which is the **Software Console**. Selecting this toggles a window that has a default location on the screen. **Software Console** affects the physical adjustments of conventional fixtures. Here one can adjust the shutters, edge, and iris of the fixture. Full screen is the other useful selection under the **Window** and expands the window to full screen.

Finally, there are two other selections on the menu toolbar at the top of the screen; **Render** and **DMX**. **Render** has options that allow users to render still images or movies. ESP Vision uses a third party application called LuxRender with a plethora of settings that can be adjusted to create the best possible image. There is a quick-render option that basically takes a snap shot and saves it to the Vision folder. **DMX** will be covered in Chapter 4.

**Vectorworks**

ESP Vision as a stand-alone application is incredibly ineffective. Without an external controller there is no way to adjust the lights in the show. Remember that ESP Vision is not a CAD program, and it works best in conjunction with external CAD software. ESP Vision works exclusively with Vectorworks for importing show files, which is perhaps its greatest feature and what ultimately influenced my decision to purchase this program over others. Its ability to import from Vectorworks is preferable because of how simple and effective the process is.
When installing ESP Vision, users will receive a prompt asking if they would like to also install the Vectorworks plugin. Once this has been installed, access is granted to an ESP Vision Command in the Vectorworks Workspace Editor (Figure 3.7) which can then be moved over to Menus. The workspace editor can be found under Tools > Workspaces—now there is a Vision option in the Menu toolbar.

![Figure 3.7: Workspace Editor within Vectorworks](image)

Once the tool bar has been added, Vectorworks will have a new set of functions specifically for exporting to ESP Vision. Users can build and 3D model a set in Vectorworks or even take a previously modeled file and prepare it to be exported. The only downside to using an already modeled file is that the fixtures will need to be changed out.

ESP Vision comes with its own package of symbols to be used in Vectorworks. In 2D, these symbols look and behave the same as the stock Vectorworks lighting
symbols, but in 3D the symbols become simple block objects instead of 3D models with unnecessary geometric data. While they may look the same, their use is crucial for the export into ESP Vision because “Vectorworks symbols do not have any photometric or DMX information attached to them.”

Accessing these symbols is the same as accessing the Vectorworks symbols; the only difference is that they are located in the Vision Files folder rather than through Vectorworks’ Libraries.

Once the space has been modeled and textured and ESP Vision light symbols have been inserted, the next step is to populate the light data before the export. This step assigns light data, the information ESP Vision needs to recognize the fixtures to selected lights and groups them based on selection. Grouping can be helpful when switching over to ESP Vision. First select fixtures of the same type that are to be grouped together.

Under the Vision Menu (Figure 3.8) select Edit Vision Data… which will open a pop up window (Figure 3.9) to edit light data. Here is where changes can be made to the fixture(s) data before export. The Fixture Name option is how the light will be identified once in ESP Vision. This is why it is helpful to select groups of the same fixtures so different fixtures don’t get filed under the wrong folder within the “Root” menu. With only one light selected, there are real values for Universe and Channel, whereas multiple lights selected will result in Multiple Selected for the description.

---

*Figure 3.8: Vision drop down menu*

---

2 Joseph
Comments will describe any important information about that fixture. As shown in Figure 3.9 there may be additional profiles available offering different specifications.

Along the top of the dialog box are four additional tabs that can affect the final data exported to ESP Vision. Gobo, Color, and Animation Wheels tabs can be used to change stock gobo and color wheels of that fixture if for instance there is a different wheel or individual color or gobo changes. The Attachments tab has options for adding Color Scrolls, Sea Changers, and Twin Spins to conventional fixtures.
There are two more selections under the Vision menu (Figure 3.8) at the top of the screen. ESP Screen is used to assign an object as a video screen and tutorials for its use can be found online. Export ESC… is the final step in the process before moving into ESP Vision. After all the lights have been processed through Edit Vision Data… the show file can be exported. After selecting Export ESC… a window will pop up asking for a name and location to export the file. By default the file will export to the Vision Files folder. Something I have learned through trial and error is that groups in Vectorworks don’t export well; they will be missing in ESP Vision. Before exporting make sure to ungroup all scenery.

The file exported from Vectorworks as an “.esc” file can be opened in ESP Vision through the menu options File > Open. Assuming everything worked correctly the scene should appear at the center of the space. There is also new information under the ROOT menu in the scene graph, which looks similar to Figure 3.5. Everything in the scene that is not a light is grouped together and retains its .esc file name. As mentioned before, the lights are now divided into folders based on data was selected and edited. This extra step of prioritization cleans up the menu and makes navigation significantly easier.
Chapter 4. Console Hookup

ESP Vision is practically useless without an external console to control the lights within the scene. Therefore, we must connect a lighting console to the computer and it is a fairly simple process. However, students often times only have one lighting console to choose from. Luckily, the three most common lighting consoles (Ion, Hog, and grandMA) all have on Windows versions of their software that can also connect to ESP Vision. These are the three lighting consoles I used to hook up to ESP Vision, but it should be noted that although plenty of other consoles can connect as well, these three are more common in the industry. The following connection processes are for running ESP Vision and the controller on the same computer. If the computer does not have sufficient RAM or processing power it is recommended you use two separate computers.

ETC NOMAD

Up until 2014 ETC had an Offline Editor available for programming shows on a computer instead of being in front of a console. In 2014, ETC released Nomad, an offline editor that connects to a lighting network and has full control over the lights. Unlocking this functionality requires the purchase of a dongle. Without the dongle the Nomad software works the same as an offline editor. ETC is perhaps the easiest of the three different external controllers to hook up to ESP Vision and requires only a few network adjustments of the computer and Nomad within the shell settings.
First, locate your computer’s IPv4 settings (Figure 4.1) and change your IP to a custom address. The address used does not matter as long as it is a valid IP address. While changing the IP address, also change the subnet mask to 255.255.255.0. This subnet mask is crucial to the connection process between ESP Vision and Nomad.

Exit these screens and open the ETC Nomad software, select Settings (Figure 4.2), then Network on the right hand side of the Settings screen. Under the console menu (Figure 4.3) confirm that these IP and Subnet mask settings match the ones set previously. These should auto fill to the Window’s settings, but it is still a good idea to double check. The next window allows one to change the network protocols for output. ESP Vision works with ETC Nomad through ArtNet so that should be the only protocol selected. Now that the settings are in order it is time to launch the two applications. On the ETC home screen there is an Offline option and an Offline w/viz option. Make sure to select the w/viz option.
Launch ESP Vision and start a new show or open an old one. The menu toolbar has a DMX drop-down menu with a few different selections. At the top of the list is Change DMX Provider followed by five other options, whose function is not all that important to the goal and are beyond the scope of this text. After selecting Change DMX Provider a pop-up screen with a drop-down menu will appear (Figure 4.3). This is where one should change the DMX provider depending on what console is being used at the time. For ETC, select ArtNet and press OK to close the window. Now ESP Vision should be synced with Nomad.

**Hog**

Hog works differently than ETC, and it is best to start the connection process with ESP Vision rather than the controller. The network settings for the Hog connection can remain as they are, including DHCP as Hog does not connect to ESP Vision through ArtNet. Therefore, the DMX provider needs to be changed. There are two different Hog DMX providers but I have had most success with Hog Net. After selecting Hog Net within ESP Vision, a window will pop up (Figure 4.5) showing the status of the connection.
Figure 4.4 show the status information for a functioning connection. If Hog4PC is not running the status will say Locating Wholehog Network and info will say Scanning port 6600…. To establish the connection, open Hog4PC and open or start a new show. There are no settings that need to be changed within the Hog4PC application. Once your Hog Net screen sets the status to running, the connection has been established.

GrandMA

GrandMA offline editors are by far the most finicky and difficult to connect to ESP Vision. A plugin must be downloaded before even trying a connection, but it must be the correct plugin version depending on which version of grandMA you are trying to connect. Rather than guessing if the plugin version is correct, it is easiest to just download the latest version of the grandMA editors and install the latest version of the plugin. GrandMA onPC version (6.808) and grandMA2 onPC version (3.0.0.5) match ESP Vision Driver version (2.900-6.700) and can all be downloaded from http://www.malighting.com/en/support-downloads/software.html.

Network settings changes are also unnecessary for connecting to grandMA consoles. However, it is necessary to start on the console side of the connection as with Nomad. First open up grandMA1 or grandMA2 and locate the Setup key (Figure 4.5). Navigate to the MA-Net Control screen (Figure 4.6) to start the connection process. The
settings default to the necessary settings for the connection, but if these have been tampered with it is easy to correct them.

Session name, password, station name, and priority can all remain as they are. The settings that are important to the connection are Session Id and Station IP. If the connection is happening on one computer then the IP is 127.0.0.1, if it is happening between two computers, the IP should match the computer’s IP. Notice the two Station IP options in figure 4.7. The Session Id can be any number desired but needs to be remembered for the next step.

Click Create Session and close out of MA-Net Control.

After successfully loading the ESP Vision Driver a new selection will appear in the computer’s Control Panel called grandma-ESP Vision Driver (32-bit). Once selected, a window will pop up (Figure 4.6) with three different settings to adjust. MA-Net Mode is changed depending on whether grandMA or grandMA2 is being used. MA-Net and MA-Net2 match the applications respectively. Figure 4.5 is a view of the
settings from grandMA2 but the same process is completed for grandMA. Beneath MA-Net Mode is the Station IP Address, which should match whichever IP address was selected on the console. Lastly is the MA Session ID selection. This number must match the Session ID set on the console. Press OK to accept the settings and close the window. Finally, open ESP Vision and change the DMX provider to grandMA.

Connection Checklists

ETC:

- Change computer subnet mask to 255.255.255.0
- Open ETC Nomad or start up console
- Click “Settings”
- Click “Network” tab
- Confirm IP settings match computers IP settings
- Deselect any other network protocols and select ArtNet as the default protocol
- Click “Okay” and confirm restart of program
- Open Nomad and select “w/Vis” or start console and continue as “Primary”
- Open ESP Vision
- Select DMX
- Select “DMX Provider”
- Change to “ArtNet”
- Press “Okay” to finish

Hog:

- Open ESP Vision
- Select DMX
- Select “DMX Provider”
- Change to “Hog Net”
- Press “Okay”
- Open Hog4PC or start up console
- Start a new show or open an existing show
- View Hog Net dialog box to confirm connection status as “Running”

GrandMA2:

- Confirm correct plug-in is loaded onto computer
- Open grandMA2 or start up console
- Select “Setup”
- Navigate to “Ma-Net Control”
- Confirm IP settings: 127.0.0.1 for single computer connection or valid IP address for multi computer connection
- Click “Start”
• Click “Control Panel”

• Click “grandMA 3D Plugin”

• Confirm settings match “Ma-Net” settings on console or offline editor

• Select “Okay”

• Open ESP Vision

• Select DMX

• Select “DMX Provider”

• Change to “grandMA”

• Press “Okay”
Chapter 5. My Class

The pinnacle of all of this work is to set up a special topics class in programming and introduce undergraduate students to new lighting consoles. Its inception came from my final years as an undergraduate. I was excited and ready to learn as much about my particular niche of the industry, lighting console training, but found this difficult due to the lack of resources my school had. This meant if there was software I wanted to learn to use, I would need to download it and teach myself how to use it.

Around this same time I was an active member in our school’s United States Institute for Theatre Technology (USITT) group and had been to a couple USITT conferences. It is at these conferences that I got a look at how companies such as ETC, grandMA, and Hog were showcasing their equipment on the stage expo floor--they were using visualizers. I was hooked and began looking into different visualizers and their uses, particularly those I could use for free. Capture was the best choice then because it was free and connected easily with my school’s Ion.

In the fall of 2013, I had the idea of teaching an advanced programming class using offline editors and a visualizer. After speaking with Professor Alan McEwen, I was tasked with finding the equipment and visualizer needed. As mentioned before, we decided to use ESP Vision because it fit best within the necessary criteria of cost, effectiveness, and continuity into the future of the program. The next task was to develop the course and get it prepared for spring of 2015.

WVU School of Theatre & Dance has a light board programming class taught by Professor McEwen. The class covers programming techniques on the ETC Ion and the ETC Element. Much of the focus of the class is on the fundamentals of programming for
theatre with a handful of advanced techniques like time code, pixel mapping, and magic sheets. The advanced programming class would recap some of the advanced topics from the ETC Ion and focus on ETC’s network software and equipment including: remote device management, gateways, and ETC Concert. Along with ETC equipment, the class would also cover programming with grandMA2 and Hog4, both using their offline clients.

There were limitations to consider when setting up the class, though. First, there is no physical console in front of the student. This proves difficult for the student to grasp some concepts like certain keysets and how they function. There is also no encoder wheel, only an encoder icon that responds to click and drag, which makes multi-parameter fixtures a little difficult to use. Software updates can also be difficult with the offline editors. For example, we will examine a student who uses brand new fixtures within their show. Unfortunately, the fixtures were new enough that their fixture profile was not on the current ETC Nomad software. There is a library update available for the ETC Eos Family including the new fixtures, but it could not be applied to Nomad. Touch screen monitors are used extensively across all three platforms: Hog, grandMA, and ETC. Aside from their obvious convenience, they are most useful when programming live events, and their absence is not a terrible set back.

ESP Vision also brings a limitation to the table that was initially unforeseen. There are certain fixture profiles that are available in the ESP Vision library that don’t have a matching profile on the consoles. The solution was easy enough and opened up a learning opportunity. As a work around, the students edited the fixture profile on the
console to match those used in ESP Vision. In some circumstances the student needed to change out the fixtures.

This advanced programing class was broken up into four sections: Vectorworks and ESP Vision, ETC, grandMA2, and Hog4. ETC was chosen as the second section because of the students’ previous course work with Professor McEwen. If the students were struggling with Vectorworks or ESP Vision, we could afford extra time with those learners on that section. Each of the three console sections end with a light show project. The light show is displayed for the class and consists of two songs from the band or musical that showcases programming techniques learned in the previous weeks.

The first few weeks were spent modeling a show in Vectorworks. This show needed to be either a musical or a stage setting for a band and needed to include some form of backdrop or walls and truss work. I pushed the students to create custom textures in Vectorworks and test their own limits in modeling so we could observe what did and didn’t work when exporting to ESP Vision. The scenes created during this time will be used for the three section projects, but the final requires students to program a new show. This new show will be the opposite of their first show to help exercise different and effective programming techniques between theatre and music concerts. Examples of student works can be found in Appendix IV.
Chapter 6. The Classroom

Most theatre programs in universities have a classroom that would be appropriate for the set up I am about to break down. For some, this may be a CAD lab, and for others this may be a multi-purpose room that is used by the design/technical students and faculty. Whatever the case, the room needs to have limited access. This doesn’t mean locked down so only a few may enter; but it should be a classroom that can keep out the general population that may be passing through the building for class, because some of this equipment is expensive.

Before a visualizer is purchased, the set-up will need a computer or computers that can handle the graphics and processing required by the chosen visualizer. There are really just three requirements that need to be considered as far as the visualizer is concerned: processing power, graphics card, and RAM. What came as a shock is the difference in minimum system requirements for each different visualizer. For instance, ESP Vision recommends a 3.0 GHZ processor, 1 GB of RAM, and a graphics card with at least 256MB of video memory. Even ESP Vision’s recommendations for better graphics cards fall short compared to the other manufacturers minimum system requirements. WYSIWYG recommends only a 2.0 GHZ processor, 4 GB of RAM, and a graphics card with at least 1GB of video memory. Capture claims any computer currently on the market should be sufficient to run the program but recommends a graphics card with, “a PassMark G3D mark of 2000 or more”\textsuperscript{3}. As a comparison, the graphics cards from ESP Vision fall a thousand points short of that mark on the same

\textsuperscript{3} qtd. on web Frequently Asked Questions
chart (http://www.videocardbenchmark.net/high_end_gpus.html). LightConverse gives no minimum processing speed, 2GB of RAM, and recommendations for graphics cards based off of their own benchmark standards. If the goal is to establish a long-term system, then I recommend purchasing a computer that exceeds the minimum requirements. Ideally the computer would possess 3.0+ GHZ processor, 4+ GB of RAM, and a gaming-style graphics card. More information about minimum system requirements and recommended graphics cards can be found in Appendix III.

WVU School of Theatre & Dance has a CAD lab with around a dozen computers both Mac and PC. This was a suitable location for the class setup, but we decided to get a new computer with a better graphics card that could be dedicated to the visualization software. Instead of building a desktop from scratch we found an Alienware gaming desktop that was affordable. Our particular model is the Alienware X51-R2 (Figure 6.1) and met or exceeded the requirements recommended by visualizer companies.

Monitors may seem like an obvious when setting up a computer but what can be overlooked is how many monitors the graphics card can support. Most computers on the market can support up to two monitors and if you are using a two-computer set-up, then there is no concern. However, if you intend on using only one computer to program and visualize, I recommend having a graphics card that can support up to three monitors. This is important because it allows one monitor to be dedicated to the visualizer and the other two for the offline editor screens. In the
classroom, having the option to use a projector can come in handy as well, especially when viewing the final projects.

A standard mouse with a wheel is more than sufficient for this application. The standard keyboard is difficult to use with the offline editors as there are too many keys on a console to replicate on the standard keyboard. It then requires the use of control, alt, and shift keys or a combination of them to modify the letter or number keys. However, there are USB programmable keyboards that are popular among lighting programmers. A popular brand is X-keys, but these are fairly expensive. They come in a variety of shapes and sizes and are also backlit. With some research, I found there other programmable keyboards on the market that are more affordable. I found a company called Cherry that has a variety of keyboards that are customizable and in a row and column format (Figure 6.2).

This style of keyboard is important to the success of the class. Being proficient at programming on a lighting console requires an understanding of the syntax but also, and possibly more importantly, muscle memory for the key layout. By replicating the same layout it allows the user to gain muscle memory which can translate to a real console down the road. It also allows for custom key labels and/or colors and keeps the user from needing to remember tedious key combinations on a standard keyboard.

The class allowed students to first learn the syntax and basic functions of each lighting console and later get their hands on the physical console to see how it compares.
Eventually, the students were able to use the real version of each console through a variety of different outlets. We have an ETC Ion in the theatre that students have access to almost any time they want. In another venue at West Virginia University there is a Road Hog system that we could access via a fieldtrip. USITT Light Lab 2015 had a grandMA2 system as well as a Hog4 and an Eos that the students had access to, if only briefly.

The physical arrangement of the workspace can quickly become congested, particularly if two computers are being used. As outlined previously, the set-up at the WVU School of Theatre & Dance is for both programming and visualization to be operated on the same computer. If two computers are being used, it becomes aggravating to switch between two keyboards, two mice, and a programmable keyboard if available. The best solution to this would be some form of USB switch that toggles between the two different computers.

Monitors become an issue, as well. Three monitors take up a significant amount of room on a desk and laying them out side by side does not create a very efficient view. A monitor tower (Figure 6.3) is perhaps the most helpful piece of hardware on the table. It creates an attractive format that is more conducive to the workflow between monitors. As mentioned previously, our computer can only output video to three monitors. The fourth monitor is for a light console when available. When final projects are due there is a VGA switch, which reroutes the signal to the classroom projector for a larger viewing.

Aside from the visualizer and any custom software that comes with it there are a few other applications that are helpful to have on the PC. ESP Vision works exclusively
with Vectorworks for creating shows so it is great to have Vectorworks on the same PC. If there are any problems in the show file they can quickly be changed and exported rather than having to get on a separate computer and transfer via a USB jump drive, though that is an option. If a programmable keyboard is being used it will probably also include software to be loaded on the PC.

Something that can be very helpful is finding a shortcut to switch IP settings when needed. For our application, it was most effective to create a .cmd file that prompted the user to toggle between DHCP settings and our custom IP address. This keeps the student from getting lost in the settings menus and allows them to get to work more quickly. Reference Appendix I for riser diagram of full system setup.

Figure 6.4: Full set up at West Virginia University including: Alienware X51-R2, four monitors, monitor tower, Cherry Rows and Columns programmable keyboard, and VGA splitter.
Chapter 7. Comparisons

Using the visualizer cannot replace time in the actual venue when designing a show, nor can using an offline editor take the place of being seated in front of the actual console. The focus of this text is to explore how educators can use visualizers as a teaching tool. When linked up with a console offline editor, we can teach students how to program on different consoles. We can also use massive fixture libraries to better understand how moving fixtures behave in comparison to one another. Typically the visualizer is used as a tool for conveying a lighting design to other people instead of using a static image and colorful words. Its other use, as I have discussed, is programming a show before getting into the venue to allow the designer more time to refine the looks rather than build from scratch.

The following will be three different groups of comparisons of the visualizer and keyboard layouts. First we will take a look at how ESP Vision works as conceptual previsualization by comparing renderings to realized show photos. Second, there will be an accuracy comparison showing how pan, tilt, color, gobo, and shutter parameters translate from ESP Vision to the physical space. Finally, there will be a hardware comparison showing customizable keyboard formats next to the console face panel they are replicating.
Show Previsualization

When choosing a visualizer that best suits the user’s needs, it is important to take image quality into account. While ESP Vision’s final picture is not as photo realistic as WYSIWYG, the decision to go with Vision was based on live performance more than render quality. That being said, with enough time the user can visualize fairly accurate representation of what a scene will look like on stage. Here is a marketing photo from ESP Visions website followed by a couple of quickly made comparison photos from our recent production of *The Glass Menagerie*.

![ESP Vision Marketing Photo](ESP Vision: Vision Pro)

Figure 7.1: ESP Vision Marketing Photo (ESP Vision: Vision Pro)
Figure 7.2: The Glass Menagerie “Laura Alone” rendering

Figure 7.3: The Glass Menagerie “Laura Alone”
Figure 7.4: The Glass Menagerie “Toms and Amanda” rendering

Figure 7.5: The Glass Menagerie “Toms Lament”
Accuracy Comparison

In Vectorworks, I quickly rendered some wall and a floor that fit the dimensions of the Gladys G. Davis Theatre at WVU’s Creative Arts Center. I added an outline of a square to the rendering for a reference to make shutter cuts and focus on that could be easily mimicked in the theatre. The following is a comparison of pan, tilt and shutter cuts using a Clay Paky Alpha Profile 700, as well as a comparison of a template and a color wheel between two colors.

Figure 7.6: Notice the blue square in the space, used as a point of reference. (Rendering)

Figure 7.7: Notice the bottom edge of the taped out square in red.
Figure 7.8: Notice the color wheel split between colors. (Rendering)

Figure 7.9: Notice the same split at the bottom of the beam.
Keyboard/ Face Panel Formatting

ETC Ion

Figure 7.10: Ion keyboard layout

Figure 7.11: Ion face panel
High End Hog4

Figure 7.12: Hog4 keyboard layout

Figure 7.13: Hog4 virtual face panel
Figure 7.14: GrandMA2 keyboard layout.

Figure 7.15: GrandMA2 Face panel.
Chapter 8. Final Thoughts

Through the course of this project there were many challenges to overcome. There was definitely an evolution of design and layout over time caused largely by trial and error. The initial vision was to set up a computer that could be used exclusively for visualization, whether for class work, personal training, or preprogramming a show. After choosing a visualizer that I felt best suited the needs of the project I compiled a list of necessary hardware.

Initially there was a two-computer set up, one running the visualizer and the other running the offline editors. I thought that two computers would allow better rendering and little-to-no lag between commands and live visualization. After spending some time speaking with tech support, I discovered two computers were not necessary and ended up switching to one. In most cases, two computers allow the user to have four individual monitors; however, this also means two mice and two keyboards, which can clutter the workspace. A work-around for the extra keyboard and mouse could be a USB switch or specialized software. This particular computer allowed the use of three separate monitors, so I ended up dropping the other computer.

Before setting up the class I already knew of a couple students who would be interested in taking the class. These students had previous training on the ETC Ion, therefore, were already familiar with fundamental programming concepts. The first console covered in the course was the Ion to review the basics. This particular class works best with a small group due to the fact that everyone needs to be around the same computer. Also, the single computer needs to be shared among the group to complete class projects. This class is also not for everyone, the success of the class is largely
dependent on the student. The ideal student should be open minded, creative, and ambitious in order to take full advantage of the class.

Finally, the longevity of this project is important to me. Its immediate effects are not only beneficial to myself and my class, but other students who have wandered into the CAD lab and used it over the past year. The goal of this project is to keep benefiting students well into the future, but for this to be possible the resources need to be easily accessible to the students. In that regard this project was a success, considering over the past year students outside of the special topics class have used the application. Other schools may struggle with the resources for this exact set up, but it is important to make this information available to the students. Ambitious students should not be limited by a lack of resources; rather, challenged to find new ways to achieve their goals.
Appendix I: Riser Diagram

Above is a simple riser diagram of the set-up at WVU School of Theatre & Dance. The Ethernet switch capacity is dependent on desired network connections to the system and can be changed accordingly. This drawing is not to scale.
Appendix II: Glossary of Terms

ACN- “Architecture for Control Networks (ACN) is a network control protocol which is used in the entertainment industry.”

ART-NET- “Art-Net is an easy-to-use way of transporting DMX and RDM data over closed, private Ethernet networks using UDP and IP.” (Huntington, 252)

AutoCAD- A CAD program used by theatre practitioners primarily for drafting scenery.

CAD- Computer Aided Drafting

DirectX- “DirectX is a set of low-level Application Programming Interfaces (APIs) that provides Windows programs with high-performance hardware-accelerated multimedia support.”

DMX512- “DMX512 is a standard that describes a method of digital data transmission between controllers and lighting equipment and accessories.”

Fixture Profile- The DMX512 channel layout for a particular fixture.

Lightwright- “Lightwright is a unique cross between a spreadsheet and a database, designed specifically to manage professional lighting design paperwork.”

4 qtd from Owaits

5 qtd from Microsoft Windows XP – What is DirectX

6 qtd from USITT

7 qtd from Lightwright
Offline Editor- A virtual version of a lighting console which allows the user to make changes to and save show information

OpenGL- “OpenGL (Open Graphics Library) is a cross-language, multi-platform application programming interface (API) for rendering 2D and 3D vector graphics.”  

Remote Device Management-““RDM”, is an official ANSI standard, which allows compatible devices to communicate over DMX cabling.”

Vectorworks- A CAD program used by theatre practitioners for drafting scenery and light plots.

---

8 qtd from OpenGL Overview

9 qtd from RDM Protocol Home page
Appendix III: System Requirements

WYSIWYG

Minimum System Requirements:10

- Intel Core i3 or compatible processor, 2.0 GHZ, or better
- PC running Windows Vista with Service Pack 2, Windows 7 with Service Pack 1 or Windows 8.0/8.1 - OR - MAC with Intel Processor platforms running Windows Vista with Service Pack 2, Windows 7 with Service Pack 1 or Windows 8.0/8.1 via Bootcamp (recommended), or Parallels Virtual Machine; please note that WYSIWYG installed on a Parallels Virtual Machine may run with limited graphical functionality
- 4GB RAM
- 1GB free disk space (NTFS file system)
- OpenGL-accelerated “gaming-level” video card, with 1GB (or better) video memory and OpenGL 2.1 (minimum) support; an integrated video card may be used, but this is not recommended.
- Free USB port

For more information visit http://cast-soft.com/content/wysiwyg-system-requirements

Capture Sweden

Minimum System Requirements:

“Apart from our current minimum operating system requirements, Capture should run fine on any standard desktop or laptop computer currently available for purchase. However, bear in mind that visualization software is very demanding on the video card. As a good referenced point for the capabilities of your hardware you may find PassMask Software's video card benchmark pages for high end GPU’s useful. A PassMark G3D mark of 2000 or more could be considered a general minimum performance requirement for Capture, but in practice the actual requirements will vary a lot depending on how you use Capture.”11

10 Bulleted points gathered from http://cast-soft.com

11 qtd from Frequently Asked Questions
LightConverse

Minimum System Requirements: 12

- CPU AMD Athlon or Intel Core
- AMD/nVidia/Intel 3D videocard (Shader-Model 3.0 compatible)
- Display resolution greater or equal to 1366*854
- 2GB RAM
- OS Windows-XP x86 SP2
- HDD free space 700 Mb
- USB port

GrandMA 3D

Minimum System Requirements: 13

- IBM compatible PC:
- Dual Core Processor with 2.4 GHZ CPU (SSE2) (recommended: Intel i7)
- >= 2 GB RAM (recommended: 8 GB RAM)
- Hard-drive >= 32 GB (recommended type: SSD)
- Network card 100/1000 TX/T
- USB 2.0 & USB 3.0
- 3D graphics card with hardware acceleration and
- 1024 MB graphic-RAM and (recommended: 3 GB graphic-RAM)
- Vertex Shader Version >= 3.0
- Pixel Shader Version >= 3.0
- Resolution >= 1024x768 (recommended: 1920x1080)
- Operating systems: Microsoft Windows 7 (recommended: Home Premium or better), Microsoft Windows 8
- All with Admin-right

12 Bulleted points gathered from http://www.lightconverse.net/lc/download-1.htm

Appendix IV: Student Work

With the permission of my students here are some stills of the classwork turned in for class projects.

Figure IV.1: Daniel del Busto’s work from *The Who’s Tommy*
Figure IV.2: Rachel Lake’s work from a concert of Florence + The Machine songs

Figure IV.3: Savannah Yost’s work from *American Idiot*
Appendix V

The following is a sample syllabus for the advance programming class taught in spring of 2015.

THEAT 429 Adv. Programming
T-Thr: 8am-9:15
Spring 2015
Instructor: Taylor
Office Hours: By Appointment
(also check the cad lab)
Office: CAC, Grad office
Phone: 7852012787
Email: the.joshua.taylor@gmail.com

Required Supplies: Flash Drive

Non-required texts for further reading:

The Automated Lighting Programmer’s Handbook- Brad Schiller

“If you are a person with a disability and anticipate needing any type of accommodation in order to participate in this class. Please advise me and make appropriate arrangement with Disability Services (293-6700).”

Attendance will be taken for all class sessions.

Absences: Attendance is required. Only 2 (two) unexcused absences will be permitted. At the instructor’s discretion, additional absences for medical or massive personal welfare issues might be excused. Students who miss excessive classes for such reasons will be asked to withdraw from the course to protect the integrity of the learning process. Please note: Involvement in theatre (and in any extracurricular activity) requires students to be organized.

Additional unexcused absences will result in lowering the final class grade by one half grade; A+ to A-, A- to B, etc. for each absence thereafter.

Tardiness: All classes start at the specified time. Students should be here and ready by that time. Arriving late for two classes will be considered equal to one unexcused absence and could affect your grade as described above.
Some of the Learned Outcomes from this course:

- Develop the skills for pre programming a show on any console
- Increase and improve programming skills on the ETC ION
- Develop programming skills on the High End Hog systems
- Develop programming skills on the GrandMA2 Systems
- Increase general lighting desk intuition
- Increase general design and programming skills
- Grading and Evaluation:

  A  90-100%
  B  80-89%
  C  70-79%
  D  60-69%
  F  0-59%

Project breakdown:

- Vectorworks Project/Show build  25
- Ion Show                        50
- Hog Show                       50
- MA Show                        50
- ION Project (Programming)      25
- Hog Project (Programming)      25
- MA Project (Programming)       25
- Final                          (50)
- Total:                        300

Projects are due at the **start** of the class indicated. Projects (not in class) turned in late will be assessed a penalty of 5 points per day. This class will be part lecture and part seminar. There will also be ‘hands on’ work in the spaces when available. Please note; the schedule may be subject to change. However any change will be announced in class or via MIX giving time for adjustment.

“The West Virginia University community is committed to creating and fostering a positive learning and working environment based on open communication, mutual respect, and inclusion.

*If you are a person with a disability and anticipate needing any type of accommodation in order to participate in this class, please advise me and make appropriate arrangements with the Office of Disability Services (293-6700). For more information on West Virginia University's Diversity, Equity, and Inclusion initiatives, please see http://diversity.wvu.edu.*

[Adopted 2-11-2013]
Schedule:

13-Jan  Intro to Vectorworks  Pick a show
15-Jan  Work Day  Create show
20-Jan  Vision
22-Jan  Import/ Focus
27-Jan  Finalize show/ Wring Out  Show choice due
29-Jan  Intro to ETC
3-Feb  ETC
5-Feb  ETC
10-Feb  ETC
12-Feb  ETC
17-Feb  ETC Showcase  ETC Project Duc
19-Feb  Intro to MA
24-Feb  MA
26-Feb  MA
3-Mar  MA
5-Mar  MA
10-Mar  MA
12-Mar  MA Showcase  MA Project Due
17-Mar  USITT
19-Mar  USITT
24-Mar  Spring Break
26-Mar  Spring Break
31-Mar  Intro to Hog
2-Apr  Hog
7-Apr  Hog
9-Apr  Hog
14-Apr  Hog
16-Apr  Hog
21-Apr  Hog Showcase  Hog Project Due
23- Apr  Work Day  Pick show for Final
28-Apr  Work Day
30-Apr  Work Day
6-May  Final
Bibliography


*LIGHTCONVERSE 3D SHOW PLATFORM main site.* LIGHTCONVERSE. n.d. Web. 20 April 2015.

PassMark Software – Video Card (GPU) Benchmarks – High End Video Cards.


Smith, Steven. WYG. Message to Joshua Taylor. E-mail. 16 February 2015.
Works Cited


Huntington, John. *SHOW NETWORKS & CONTROL SYSTEMS.*


21 April 2015.


20 April 2015.

Owaits. *Architecture for Control Networks (ACN) – Home https://acn.codeplex.com/.*


