MBOX® MINI

USER MANUAL

Mbox® Mini User Manual

Version as of: December 14, 2014

PRG part number: 02.9800.0004.39

Production Resource Group
Dallas Office
8617 Ambassador Row, Suite 120
Dallas, Texas 75247
www.prg.com

TABLE OF CONTENTS

Chapter 3. Configuration

Chapter 4. Operation

Chapter 6. Video Input

Chapter 7. Content Creation

Appendix A. Networking For Mbox

Appendix B. Parameter Mapping

REVISION HISTORY

This manual has been revised as follows:

Version	Release Date	Notes
02.9800.0004.38	May 9, 2014	Initial release.
02.9800.0004.39	December 14, 2014	Updated to software version 3.9

Notes

INTRODUCTION

About This Manual

This manual provides necessary information regarding product safety, installation, and operation for the following PRG product:

+ Mbox® Mini

This manual applies to software version 3.9.

Familiarizing yourself with this information will help you get the most out of your PRG product.

> ⚠ **WARNING:** It is important to read ALL accompanying safety and installation instructions to avoid damage to the product and potential injury to yourself or others.

Important Note About Consoles!

The Mbox Media Server is designed to work with moving-light consoles. Consequently, most of its features are not compatible with conventional consoles. Mbox Designer v3.9 works with the following moving-light consoles: PRG V676®, V476® and Virtuoso® consoles, MA Lighting grandMA 1 and 2 consoles, High End Systems Hog® 3, Hog® 4, or iPC consoles (running Hog 3 or 4 software only), the ChamSys MagicQ console, and the Martin Maxxyz™ and M series consoles. Mbox Designer v3.9 is not compatible with High End Systems Hog® 2 or iPC consoles running Hog 2 software. Neither PRG nor High End Systems will support the use of Mbox Designer v3.9 on any console running Hog 2 software.

In general, if you are using Mbox with any console other than Vx76, Virtuoso or grandMA, it is a good idea to contact PRG for more information.

Additional Documentation

For more information, refer to the following manuals:

+ Mbox® Designer User Manual (02.9800.0001.xx)
+ Mbox® Director User Manual (02.9800.0002.xx)
+ Mbox® Studio User Manual (02.9800.0003.xx)
+ Mbox® Remote User Manual (02.9800.0005.xx)
+ PRG Lighting Systems Networking Guide (02.3004.1000.0)
+ Vx76 Software User Manual (02.9814.0001.xx)

For more information about DMX512 and sACN protocols, refer to the following documents available from the American National Standards Institute (ANSI) at www.ansi.org:

+ ANSI E1.11 - 2008 (R2013)
  Entertainment Technology - USITT DMX512-A, Asynchronous Serial Digital Data Transmission Standard for Controlling Lighting Equipment and Accessories
+ ANSI E1.31 - 2009
  Entertainment Technology – Lightweight streaming protocol for transport of DMX512 using ACN

The above documents are also available in electronic format free to PLASA members at www.plasa.org

For more information about Art-Net, refer to the following document available from Artistic Licence Engineering at www.artisticlicence.com:

+ Specification for the Art-Net Ethernet Protocol

Training Videos

Mbox Media Server Training Videos are available on the PRG website. The following is a list of videos that apply to Mbox Mini:

+ Patching: http://www.prg.com/mbox-media-server-training-video-4/
+ CITP - Streaming Feedback & Thumbnails: http://www.prg.com/mbox-media-server-training-video-6/
+ Daemon Application: http://www.prg.com/mbox-media-server-training-video-7/
+ Media Server Concepts: http://www.prg.com/mbox-media-server-training-video-10/
+ Play Modes: http://www.prg.com/mbox-media-server-training-video-11/
+ Transitions: http://www.prg.com/mbox-media-server-training-video-13/
+ Texture Effects: http://www.prg.com/mbox-media-server-training-video-17/
+ 3D Functionality: http://www.prg.com/mbox-media-server-training-video-18/
+ Shutters & Keystone: http://www.prg.com/mbox-media-server-training-video-19/
+ Codes, Storage, and Media Types: http://www.prg.com/mbox-media-server-training-video-20/
+ Pixel Remapping (9 videos): http://www.prg.com/mbox-training-videos/

Direct links to the videos are also provided throughout this manual, where applicable.

Customer Service

For technical assistance, contact the PRG International Service Center or contact your nearest PRG office. Contact information for all PRG office locations can be found on our website at: www.prg.com/about-us/locations/

PRG Dallas (International Service)

8617 Ambassador Row, Suite 120
Dallas, Texas 75247 USA
Phone: 214.630.1963
Fax: 214.630.5867
Service Fax: 214.638.2125
Service Email: orders@prg.com

For Mbox support, please contact: mboxsupport@prg.com

For additional resources and documentation, please visit our website at: www.prg.com

1.

OVERVIEW

This chapter provides an overview of Mbox features and operations.

+ GENERAL OVERVIEW

+ FIXTURE DESCRIPTIONS

+ OPERATING MODES

GENERAL OVERVIEW

Features

Mbox Mini allows advanced video playback and effects to be integrated into an automated lighting system. Powerful real time rendering, built-in effects, and interactive 3D objects provide tools for combining pre-recorded movies and still images to create visual imagery, which can be displayed by any projector, LED wall, plasma screen or LED lighting fixture.

The Mbox Mini software can be programmed and operated from a-based moving light console, allowing ease of coordination between movie/image effects and lighting cues. Intricate Mbox sequences can be generated instantly on site without the need for offline video processing, rendering or editing. Custom media files are easy to add to the server for playback on demand.

The Mbox Mini software runs on an Apple® Mac® computer controlled by Art-Net or sACN. All movies, still images and 3D objects are stored directly on the hard drive, allowing instant access to any file.

Mbox Mini Feature List
+ Two operating modes: Single Output and Panoramic Wide.
+ Accepts live video input with optional PCIe cards, Blackmagic UltraStudio Thunderbolt, and USB/Firewire devices.
+ 1080p HD Video playback. (Playback resolutions are affected by the capability of the computer hardware.)
+ Up to four interactive, scalable layers that can be configured as backgrounds or 3D objects.
+ Digital gobos that can be used as backgrounds, masks, or for advanced layer blending effects.
+ Pixel mapping.
+ Mapping of textures (movies or still images) to 3D objects and digital gobos.
+ Real-time crossfades and transitions between movies or images on the same layer.
+ Effects: 1 on Master and 1 on each layer. Over 100 Effects that can be used in combination on any layer.
+ User-configurable aspect ratio and output resolution control.
+ Control by Art-Net or Mbox Director.
+ Stores up to 65,535 movies and still images.
+ Use up to 255 3D objects.

Limits
Mbox Mini software has the following limits:
+ Content is limited to 1920x1200.
+ Output is limited to 4.2 million total pixels.
+ Pixel mapping output is limited to 3072 pixels.

Concepts of Operation

Basic Function

The basic function of Mbox is to control dynamic media - such as movie clips, still images and 3D objects - during a stage show or other performance. The resulting visual imagery is output to a display device such as a projector, LED wall, plasma screen or LED lighting fixture. Built-in features allow the media to be manipulated "on the fly" so that changes can be made during the actual performance and immediately seen on stage. This eliminates the need to modify media clips in advance, which can take considerable time. Mbox is like a combination of After Effects, Photoshop, Maya, and Final Cut Pro which can be used in real-time to project custom imagery.

Control

The Mbox Mini software can be controlled by either a lighting console or a Mac computer running OS X 10.6.7 or greater and Mbox Director software. In either case, the console or computer can be used to configure and control all aspects of Mbox media content and playback parameters.

+ **Console** - Since a lighting console would typically be used in a stage production, this method is a convenient and logical way to consolidate all stage control elements into a single device.

+ **Mbox Director** - Mbox Director is a software application that allows control of Mbox Mini from either another OS X machine or optionally from the same computer running Mbox Mini depending on the required performance of the Mbox Mini software.

This manual assumes that a lighting console is being used to control Mbox. For instructions on using Mbox Director, refer to the separate Mbox Director User Manual (02.9800.0002.xx).

Primary Components

A typical Mbox system is made up of several different components that work together to achieve the final result. The following simplified diagram illustrates how the components work together:
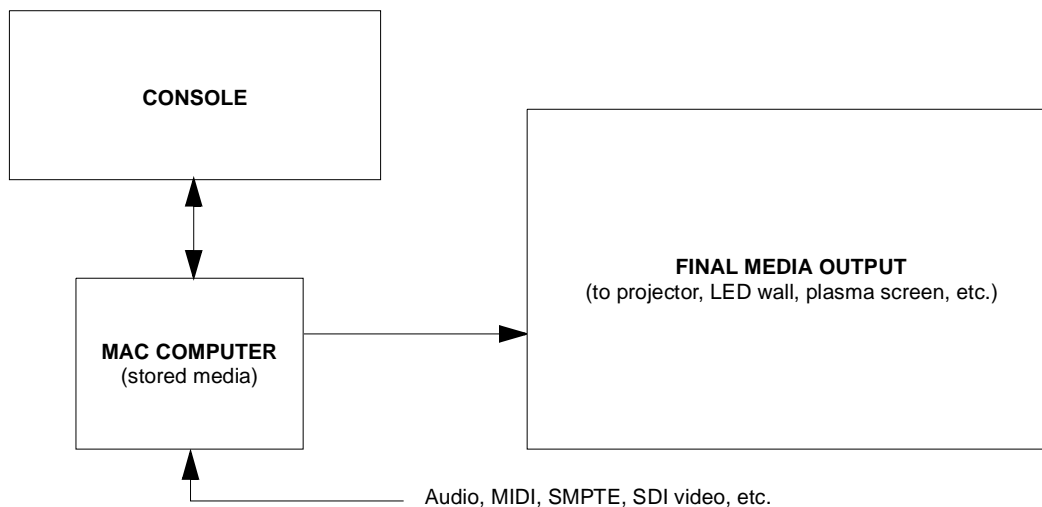
Figure 1-1: Simplified Component Diagram

Mbox Mini is a software product that requires a suitable Mac computer to achieve desired performance levels.

The following items should be considered:

+ Video Card

+ Hard Drive Configuration (RAID, SSD)

+ Number of Processor Cores and Processor Speed

+ Capture Cards

FIXTURE DESCRIPTIONS

Mbox Environment Concept

In order to get the most from Mbox Mini, it is important to understand the overall concept of the Mbox environment, which is essentially a building platform for "virtual scenery." The final "look" of this virtual scenery is comprised of multiple *layers*. These layers can be made up of backgrounds and 3D objects, all of which are controllable elements within the Mbox environment. Likewise, each of these elements has a set of controllable parameters such as color, texture, size, and rotation. When all combined, it provides an infinite number of possibilities for the virtual scenery.

At first, the interaction of these elements may seem complicated, but as they are broken down into individual components, you will find that they are similar to automated lighting parameters which you may already be familiar with. The only difference is that luminaire parameters coincide with physical hardware - e.g. a motor that controls an iris - while the Mbox parameters coincide with software elements. For example, just as you would control a luminaire's beam color from a console encoder or keypad, you can control the color of an Mbox 3D object.

Once configured, the entire Mbox environment (i.e. the virtual scenery) can be output on any compatible video device connected to the machine.

The following illustration is a virtual representation of each element within the Mbox environment. (More detailed descriptions of each element are provided on subsequent pages.)

Note:

This concept drawing shows 4 backgrounds and 4 objects. The Mbox Mini system allows up to 4 active entities at one time. For example, 4 backgrounds; 2 backgrounds and 2 objects; 3 backgrounds and 1 object; 1 background and 3 objects; etc.
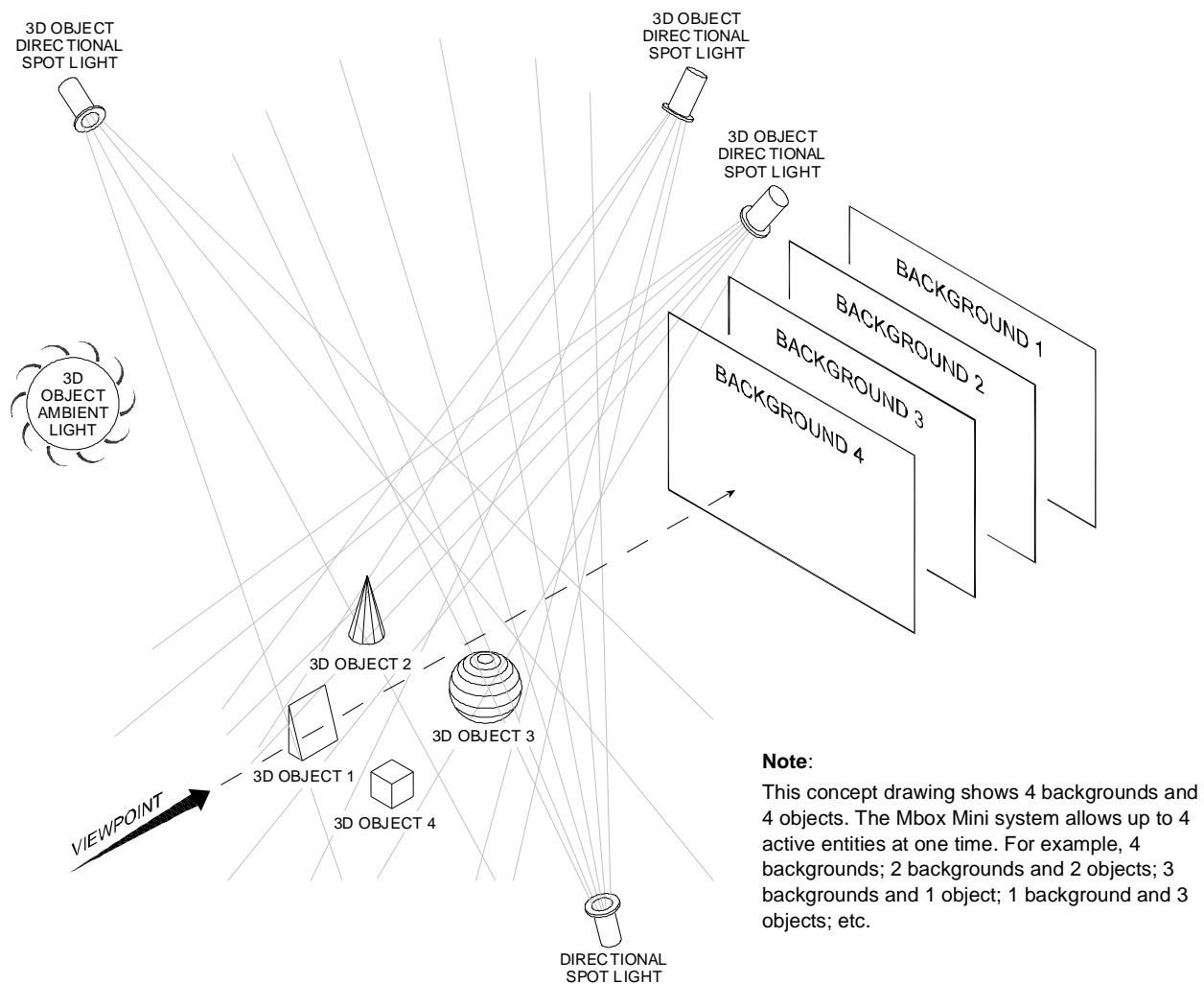
Figure 1-2: Mbox 3D Environment

Fixture and Layer Overview

The Mbox environment is based on the concept of *layers*. The system supports up to 4 active layers at a time, each having its own particular elements and parameters. The elements include a choice of still images, movies, or 3D objects. The parameters include settings such as opacity, texture, intensity, color, blending, rotation, and many more. The combination of all 4 layers results in the final "look" of the virtual scenery.

For control purposes, each layer is considered a "fixture" by the console. Some consoles may use one fixture to control 2D textures and another fixture to control 3D objects. In addition, a set of overall master settings are considered as yet another fixture. All parameters are controlled by mapping them to control channels on the console. An appendix at the end of this manual contains tables for all channels used with the Mbox Mini software.

The fixtures and their controllable parameters are outlined below. To help illustrate these concepts, an image of the corresponding "Heads Up Display" is shown. The HUDs provide feedback regarding the values (VAL) assigned to the parameters (PARAM). The HUDs will be covered in more detail later in this manual.

MASTER

+ **Fixture: Master**
    - Pixel Mapping Output Level
    - Master Control Selector
    - Effect 1 with Controls A & B
    - Master Color: Red, Green, Blue
    - Intensity
    - HUD Control
    - Master Audio Output Level

+ **Fixture: Keystone**
    - Keystone Controls 1a - 4b

+ **Fixture: Shutter (Framing)**
    - Color: Red, Green, Blue
    - Scale
    - Soft Edge
    - Shutter Controls 1a - 4b

```
Mini Master: Keystone/Shutter


        MASTER                KEYSTONE                SHUTTER
        A-001                 A-012                   A-020

    00  pmlev    255      00  key 1a   000        00  red      000
    01  ctrlSel  000      01  key 1b   000        01  green    000
    02  eff 1    000      02  key 2a   000        02  blue     000
    03  mod a    000      03  key 2b   000        03+ scale    65535
    04  mod b    000      04  key 3a   000        05  edge     000
    05  red      255      05  key 3b   000        06  sh 1a    000
    06  green    255      06  key 4a   000        07  sh 1b    000
    07  blue     255      07  key 4b   000        08  sh 2a    000
    08  dimmer   255                              09  sh 2b    000
    09  ctrl     000                              10  sh 3a    000
    10  volume   255                              11  sh 3b    000
                                                  12  sh 4a    000
                                                  13  sh 4b    000
```

Color indications:

+ Green - at default value.

+ Orange - not at default value.

LAYERS

Fixture: Texture (2D)

+ Effect 1 with Controls A & B
+ Blend Mode
+ Drawmode
+ Color: Red, Green, Blue
+ Opacity
+ Texture Folder Number
+ Texture File Number
+ Playmode
+ Playspeed
+ In-Frame
+ Out-Frame
+ Sync Stream and Sync Offset (Timecode and Layer-to-Layer)
+ Frame Blending
+ Crossfade Type and Timing

```
Mini Layers: Texture Information
      PARAM     LAYER 1  LAYER 2  LAYER 3  LAYER 4          PARAM
ADDR            A-034    A-073    A-112    A-151
00   fx 1       000      000      000      000              fx 1
01   ctl1a      000      000      000      000              ctl1a
02   ctl1b      000      000      000      000              ctl1b
03   blMod      000      000      000      000              blMod
04   dwMod      000      000      000      000              dwMod
05   red        255      255      255      255              red
06   green      255      255      255      255              green
07   blue       255      255      255      255              blue
08   opac       000      000      000      000              opac
09   txFol      000      000      000      000              txFol
10   txFil      000      000      000      000              txFil
11   plMode     000      000      000      000              plMode
12   plSpd      127      127      127      127              plSpd
13+  in Frm     00000    00000    00000    00000            in Frm
15+  outFrm     65535    65535    65535    65535            outFrm
17   sStrm      000      000      000      000              sStrm
18   sOfset     127      127      127      127              sOfset
19   fBlnd      255      255      255      255              fBlnd
20   xfType     000      000      000      000              xfType
21   xfTim      000      000      000      000              xfTim
```

Fixture: Objects (3D)

+ X and Y Position
+ Scale
+ X and Y Scale
+ X, Y, Z Rotation
+ Object File Number

```
Mini Layers: 3D Object Information
      PARAM     LAYER 1  LAYER 2  LAYER 3  LAYER 4          PARAM
ADDR            A-056    A-095    A-134    A-173
00+  x pos      32767    32767    32767    32767            x pos
02+  y pos      32767    32767    32767    32767            y pos
04+  scale      32767    32767    32767    32767            scale
06+  scaleX     32767    32767    32767    32767            scaleX
08+  scaleY     32767    32767    32767    32767            scaleY
10+  rot Z      32767    32767    32767    32767            rot Z
12+  rot X      32767    32767    32767    32767            rot X
14+  rot Y      32767    32767    32767    32767            rot Y
16   obFil      000      000      000      000              obFil
```

Color indications:

+ Green - at default value.
+ Orange - not at default value.

📖 For a complete parameter maps, refer to

Master Fixture

The Master fixture provides mastering controls for video output intensity, color, pixel mapping intensity, and audio output volume; a control channel for Mbox macros and HUDs; and one master effect.

Master Intensity

The Master Intensity parameter directly dims the overall video image on the output of the computer.

IMPORTANT! If the Master Intensity is set to zero (or a low level) you will not see any imagery on the output!

Master Color

The subtractive RGB color mixing on the Master fixture works upon the final rendered image created by compositing all layers together.

Master Audio Volume

The Audio Volume parameter masters all audio output on the server; even audio that is subject to mastering on an individual layer through the use of Opacity as Layer Volume or supplemental Layer Volume DMX controls.

Master Control Channel and Control Selector

The Control Channel and Control Selector are used to trigger macros, enable/disable features, and to bring up the Heads Up Displays (HUDs) on the Mbox output screens.

Master Effect

The effect on the Master fixture is pulled from the shader-based texture effects (1-200) that may be used on each layer. The effect is applied across the entire output context (i.e., one or both outputs depending on the mode).

Note: Some shader-based effects (e.g., transparency) applied to the Master may work differently when applied to Layers because there is nothing behind the composite image. Other effects may be incompatible with large output resolutions.

Pixel Mapping Output Level

The Pixel Mapping Output Level parameter masters the overall brightness levels of all pixel mapping output - all fixtures, all contexts, and all protocols. This control is able to dim the pixel mapping output in cases where dimming the video output would be undesirable. (The pixel mapping group controls are available for use in dimming groups of fixtures - or even individual fixtures - when dimming all fixtures is not appropriate.)

Unlike prior versions of the Mbox software, with Mbox v3.9 the Master Pixel Mapping Control Channel is used to enable/disable the overall output of the pixel mapping output signal - all protocols - and to force all "masked" fixtures to an output level of zero, thereby allowing an HTP merge with CMY fixtures.

Refer to the Mbox Remote User manual for more information about the Pixel Mapping feature.

Shutter Fixture

The Mbox software includes shutters, much like a Leko, which can be used to mask off areas of the final composited output image. The shutters can be thought of as sitting between that composited image and the virtual camera that provides the view of the image. The shutters do not only affect one layer or a group of layers, they affect the overall final image.

Shutter Color

The additive RGB color mixing on the shutter fixture may be used to apply a color other than black to the shutter blades.

Shutter Edge

The edges of all four shutters can be simultaneously adjusted from a hard edge to a soft edge. The softened edge is centered on the line of the original hard-edged shutter.

Shutter Scale

Adjustment of the scale makes it possible to create an arrangement of the shutter blades and then reduce the size of the shutter arrangement. Essentially all four shutter blades are being moved equally towards the center of the screen, while maintaining their relationship to each other.

▶ Training Video: Mbox Media Server Training Video Chapter 19: Shutters & Keystone

Keystone Fixture

The Mbox software includes keystone correction functionality that is controllable via either an incoming control signal (Art-Net or sACN) or directly from the Mbox Remote application. The descriptions below refer specifically to parameters that can be adjusted using Art-Net/sACN. For keystone/warp setup using Mbox Remote, refer to the separate Mbox Remote User manual.

Keystone correction may be used to remove the image distortion created when you project upon a surface that is not perpendicular to the axis of projection. The controls act upon a virtual "Keystone Rendering Surface" and allow the output image to be manipulated so as to correct the squareness of the image, its rotation, size, and linearity.

Mbox Mini provides one mode for keystone control: Corner Adjustment. In this mode, each of the four pairs of controls can be used to move the X/Y position of one corner of the keystone surface towards the center.

Note: The Art-Net/sACN controllable keystoning does not include warp functionality.

▶ Training Video: Mbox Media Server Training Video Chapter 19: Shutters & Keystone

Layers: Texture + Object

Each of the 4 content layers incorporates both Texture (2D) and Object (3D) functionality. The Texture element displays 2D elements such as still images and movies. The Object element displays 3D objects or digital gobos. The combination of the Texture and Object elements make up the final composition of each layer.

CAUTION! Even if you are not actively using 3D objects or digital gobos, there are object-related controls on each layer that can affect a texture on that layer (e.g., X & Y Scale, Z Rotation).

Texture

In addition to texture selection, the Texture controls allow further manipulation of each layer. For the most part, the Texture controls act in 2D space. Each layer can be colored individually; the position, aspect ratio, and scale can be changed; the playback mode, rate, in/out points, and frame blending can be controlled; the transition type and timing can be modified; and effects can be applied.

IMPORTANT! Mbox can transition from one texture to another on the same layer. This feature makes Mbox both powerful and easy to use. The ability to transition on the same layer speeds up programming and can help in live situations. A transition style (e.g., dissolves, wipes, bleeds) and a time for the Texture transition can be set on each layer. If a layer's transition time control is set above 0, Mbox will use the layer's current transition mode to change from one image to the next in the time designated.

Object

The Object controls allow selection and manipulation of 3D objects and digital gobos. The controls allow further manipulation of the layer in 3D space, as well as control over transitions and effects within the 3D environment. The Object controls include scaling on X and Y axes and rotations on all three axes.

Training Video: Mbox Media Server Training Video Chapter 18: 3D Functionality

Animated Gobos and Textures

The Mbox software has two forms of proprietary animated content. The first type is the Animated Gobos, which can be found in the Models folder. The second type is the Animated Textures, which can be found in the Media folder. Due to their location, these two types are selected using their respective folder.file controls. Each type has supplementary controls for their animated aspects.

Note: The Animated Gobos are 3D objects and, as a result, some of them require lighting to be visible.

Animated Gobos

Animated Gobos are specially created object files. Despite being objects, most of them are two-dimensional. But like digital gobos, their ability to be manipulated in 3D space makes them suited to be objects.

The Animated gobos included with Mbox Mini provide function for automatic content formatting (e.g., TV Screen or Fit to Screen) or Pixel Remapping (Multiscreen). Refer to "Animated Digital Gobos" on page 60 for more information on each Animated Gobo.

Animated Textures

Animated Textures are simple two-dimensional effects that provide repetitive motion effects on a layer. The two textures create horizontal, vertical, or radial lines and then apply motion to the lines. When Animated Textures are used on a layer, the Playmode, Playspeed, In Frame and Out Frame controls are used to control their various parameters. Refer to "Animated Textures" on page 62 for more details on the control parameters for the Animated Textures.

Effects and Transitions

Mbox Rendering Environment

Mbox loads each layer's raw imagery frame-by-frame onto an individual rendering context. After a frame of imagery has been loaded onto the context, applying an effect and/or transition can modify it. Transitions are applied at the final point in layer rendering as a mix between the outgoing and incoming content. All layers are then composited into a single context, from back to front, and the Master Effect can then be applied to that single context.

Effects

There are two types of Effects in Mbox - Shaders (1-200) and Object Effects (201 and above). Both the shaders and object effects are available to be used on Layers, but only the shaders work on the Master.

Shaders

When applied to a Layer, shaders only work within the original pixel boundaries of the content that has been selected on the layer. This means that a layer's shader-based effect cannot modify pixels on other layers, or beyond the original boundaries of the content. Even if the overall texture size is larger than the content, the result of the shader will only be visible on the content. Therefore, some shader-based effects (such as wobble) will appear to have an invisible cropping frame around them.

+ When applied to the Master, shaders work within the boundaries of the entire screen and will affect everything visible on the screen.

+ Shaders are written as mathematical expressions that modify the properties of individual pixels in the source content. There are three basic manipulations (with infinite possibilities) that shaders can make: color, opacity, and position.

+ Shaders can be created by anyone and compiled for use on the Mac platform. They can then be imported for use with Mbox.

Object Effects

Whereas shader-based effects work on individual pixels, Object Effects work on the entire composition of a layer. This means that Object Effects can work outside the original pixel boundaries of content, allowing them to move the content without it being cropped. Object Effects can move, rotate, crop, or strobe the content, and can also add complex openGL manipulations to the 3D aspects of the layer.

Note: Object Effects must be created by PRG's Mbox Development Team.

Transitions

One of Mbox's most powerful features is its ability to perform a timed transition from one piece of content to another on the same layer. There are two types of transitions within Mbox - Built-in and Object Dissolve. There are about 60 different built-in transitions, but only one Object Dissolve. The built-in transitions are based on shaders, just like the majority of the layer effects.

Built-In

As with the shader-based Layer effects, the built-in transitions can only work within the original pixel boundaries of the layer's content. What this means is that if the outgoing and incoming content share the exact same pixel size the transition can work seamlessly. However, if the two pieces of content are different sizes, the transition will happen in a frame whose size is defined by that of the larger piece of content. In most cases this should be fine, as the majority of the built-in transitions still work well like this.

One consideration is that the built-in transitions may not work as desired when other parameters such as position, rotation, scale, etc. are modified at the same time as the content is changed. In such cases, the other parameters will change based on the cue timing, while the transition uses the transition timing parameter. This can mean that parameter changes can precede or lag behind the content transition.

Note: Built-in transitions between different sizes of content will likely produce unexpected results when used with 3D objects.

Like Effect shaders, built-in transitions can be created by anyone and imported for use in Mbox.

Object Dissolve

The Object Dissolve transition type is special within Mbox. It is not a built-in shader-based transition, but rather it uses Mbox's ability to support two playback/rendering engines on each layer. When an Object Dissolve transition is initiated, Mbox automatically begins an opacity crossfade between the two pieces of content, just like you might perform manually using two different layers.

Because it works like a two-layer opacity crossfade, the Object Dissolve transition can seamlessly move between content of different sizes and with different values for parameters like position, scale, rotation, and so on.

OPERATING MODES

Mbox Mini Modes

Mbox Mini has two modes of operation: Mini (Single Output), and Mini Pan Wide (Panoramic Wide). The primary purpose of the operating modes is to configure the screen arrangement and required Art-Net universes.

Note: Mbox Mini modes are not compatible with Mbox Designer or Mbox Extreme profiles.

The Mbox Setup tab, covered later in this manual, is used to change the operating mode. (Refer to "Setup" on page 36). The modes function as follows:

Mini Mode (Single Output)

The Mini mode provides one video output from the server. (This is the most common mode of operation.) The output has one set of Master, Keystone, and Shutter fixtures. It can display up to 4 content layers.

Figure 1-3: Example of Single Output Mode

Mini Pan Wide Mode (Panoramic Wide)

The Mini Pan Wide mode provides two video outputs from the server. The two outputs create a single display surface and the overlap and blending between the two outputs can be adjusted. Outputs can be arranged either horizontally or vertically. The two outputs share one set of Master, Keystone, and Shutter fixtures. Any of the server's 4 possible content layers can be displayed on either screen or across the overlap with a portion on each screen.
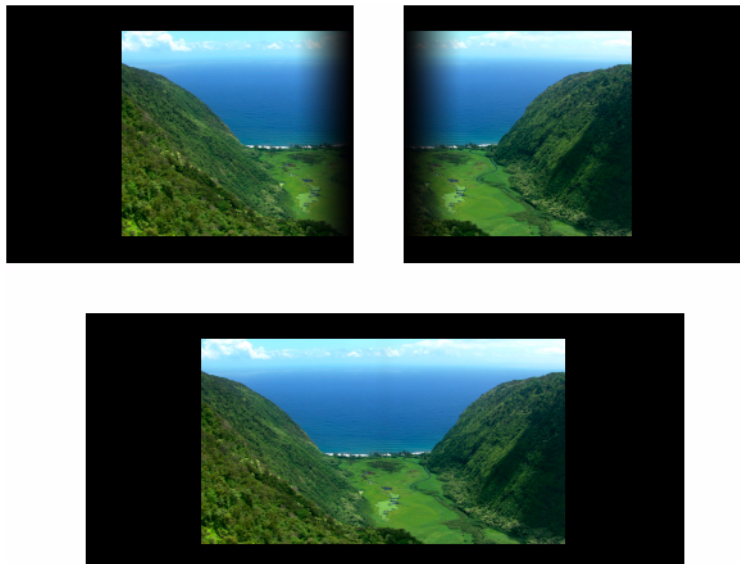

Figure 1-4: Example of Panoramic Wide Mode

2.

INSTALLATION

This chapter provides instructions for installing the Mbox Mini software and getting started.

+   GETTING STARTED

+   PATCHING FIXTURES

GETTING STARTED

Before You Start

It is strongly recommended that you install and test the Mbox software on your computer before purchasing a license!

Mbox Designer, Studio, and Mini require OSX 10.6.7 or greater to be installed on the computer and a minimum of 100MB on the Boot Drive with additional space for any custom content. The user who installs Mbox must also be an Administrator of the machine.

Note on Code Signing and Gatekeeper (for OSX 10.7, 10.8, & 10.9)

The Mbox Applications and Installers have been code-signed by Apple to denote they are both valid and have not been tampered since their creation by PRG. Since these applications cannot be delivered by way of the Mac App Store, you must make sure that your Gatekeeper settings will allow the apps to be installed and launched.

To assure a successful installation: from the Apple menu, go to -> System Preferences -> Security & Privacy and make sure that the setting for Allow apps downloaded from: has either the "Mac App Store and identified developers" button or the "Anywhere" button pressed.

Download locations:

+ The latest version of Mbox Mini can be found at: http://www.prg.com/product/mbox-mini

Installing Mbox Software

Overview

The Mbox Mini installer will place the required files in several places on your computer. You must have the necessary rights to install software on that computer (password authentication).

+ The Mbox Mini software will be stored in: /Applications/Mbox Mini
+ Utility applications will be stored in: /Applications/Mbox Utilities
+ Thumbnail and Web images will be stored in: /Library/Application Support/Mbox

The Mbox Mini installer requires the creation of an Mbox folder at the root level of your boot hard-drive, or the creation of an alternate Mbox folder plus an alias to that folder. The alias must be placed at the root level of your boot hard-drive. The installer will create the necessary alias for you.

Pre-existing Mbox folders will be renamed (rather than deleted) by the installer, allowing you to move existing content and preferences files from an older version of Mbox software to a newer version.

Content - both 2D and 3D - and preferences must be stored in the Mbox folder. The Mbox Mini installer will not place any content in the Mbox folder, but demo content is available as a free download from the Mbox webpages on the PRG website.

Thumbnails and Apache HTTP Server

As noted above, Mbox Mini places thumbnail and web images on the hard-drive of the server. These items are intended to be accessed remotely by either the Mbox Director software or any web browser. Access to the images requires the Apache HTTP Server functionality to be enabled on the Mbox server. The Mbox Mini installer will automatically enable the Apache server on your computer.

Suggested Hardware Configurations

Mbox Mini performance options:

+ **Highest performance** - Apple MacPro with 8 or more CPU cores, boot hard-drive and separate RAID or SSD drive for content, Blackmagic Decklink video capture card, ATI 4870 or 5870 graphics card, 4GB or more RAM.

+ **Medium performance** - Apple iMac with quad-core processor, boot hard-drive and separate hard-drive for content, ATI 6770 or 6790 graphics card, 4GB or more RAM, external video output adapter for connection to stage device.

+ **Portable system** - Apple MacBook Pro or Mac Mini with quad-core processor, boot hard-drive and separate hard-drive for content (external Thunderbolt drive or SSD recommended), 4GB or more RAM, external video adapter for connection to stage device.

Mbox Mini computer requirements:

+ Mac OSX 10.6.7 operating system or newer

+ Intel processor Mac computer

To install Mbox Mini:

Step   1. Download and unzip the latest package from the above link.

Step   2. Assure no applications that might interfere with the installation are running, then run the installer (e.g. all Mbox-related apps like Launcher, Daemon, or Remote). Launch the installer, read the legal and installation notes, then continue.

Step   3. At Media Path pane of the installer, choose where the Mbox Content folder will be placed, as well as what to do if an Mbox folder already exists. For a default installation, leaving the options untouched will place the Mbox Content folder at the root of the System (boot) drive of the computer.

You can use the Mbox Content Folder Destination section to cause the Mbox Content folder to be placed at an alternate location. However, there must be an item called "Mbox" at the root of the System drive, so the installer will create an alias at that location, pointing to your selected custom location.

For an upgrade installation, if a folder or alias named "Mbox" is found on the System drive, the pop-up in the Boot Mbox Folder section will be relabeled "Found" and you will have the following options:

   a. If "Rename existing" is selected on the pop-up, the existing Mbox folder or alias will be renamed as *Mbox####.old* before a fresh Mbox folder/alias is created in the same location. After the installation, you can manually move any required files into the new Mbox folder as needed.

   b. If "Merge existing" is selected on the pop-up, the installer will merge any installed files with the current installation. This option will not delete any existing content, but may change some settings and will add the 001.Sample media, default models, and plug-ins folders.

Step   4. Acknowledge your settings and then agree to the License Agreement to begin the installation.

Licensing Mbox Software

CAUTION! Always install the Mbox software before licensing!

Software-only license (eSellerate)

If you purchased a software-only license online, then you should have received a receipt and an email with your serial number. The serial number will start with the characters "PRGMBS-" for Mbox Studio or "PRGMBM-" for Mbox Mini.

To license your installation of Mbox, you must then connect the computer to the internet and launch the Mbox application. Then enter your serial number in the License area of the Mbox Setup window.
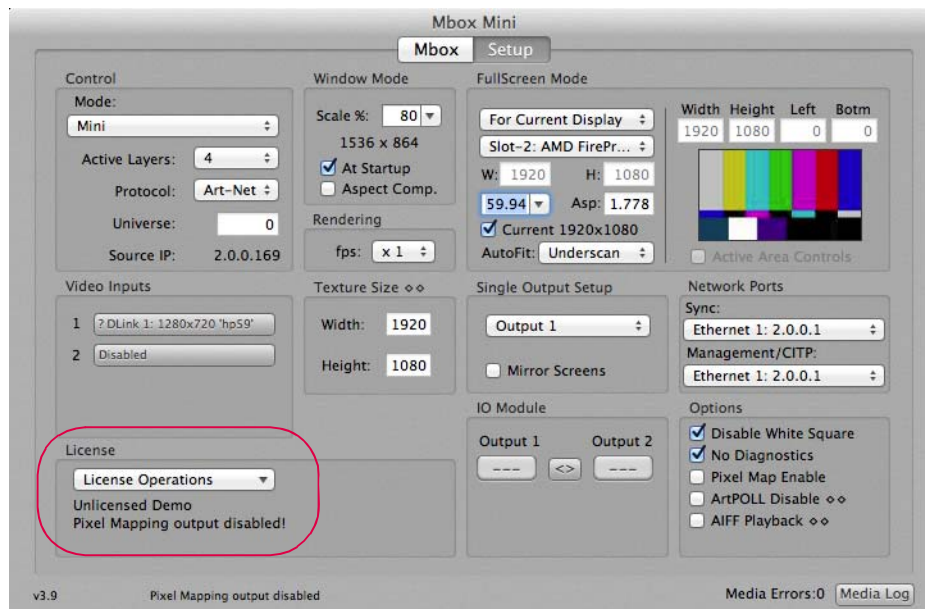
If you purchased a software-only license directly through the licensing portal built into the Mbox application, then the software should be automatically licensed at the successful completion of your purchase. You should still receive an email receipt with your serial number in case you need to re-license your computer at a later date.

Hardware License USB Key (Dongle Drivers)

When you first install the Mbox software, the installer will give you the option to install the USB license key drivers as well. If you don't have a USB license key, you don't need to install the drivers. If you purchased a USB license key at a later time, then the drivers can be downloaded and installed from the link on the previous page. Once the drivers are installed, Mbox will recognize the license key.

If the Mbox application fails to recognize the USB key at any point after the drivers are installed, please contact PRG for assistance.

Launching Mbox Mini

When launched, the Mbox Mini software will initially show a splash screen. After a brief pause, the application window will appear. During this time the software will scan and index all media. Depending on the preferences, the software may enter Fullscreen mode. In Fullscreen mode, if no control signal is being received the output screen may be black. This is normal and you can exit Fullscreen mode at any time (see below).

While in Window mode, the application window and its user interface will be shown. This window contains two tabs: Mbox and Setup as shown in this sample screen below:
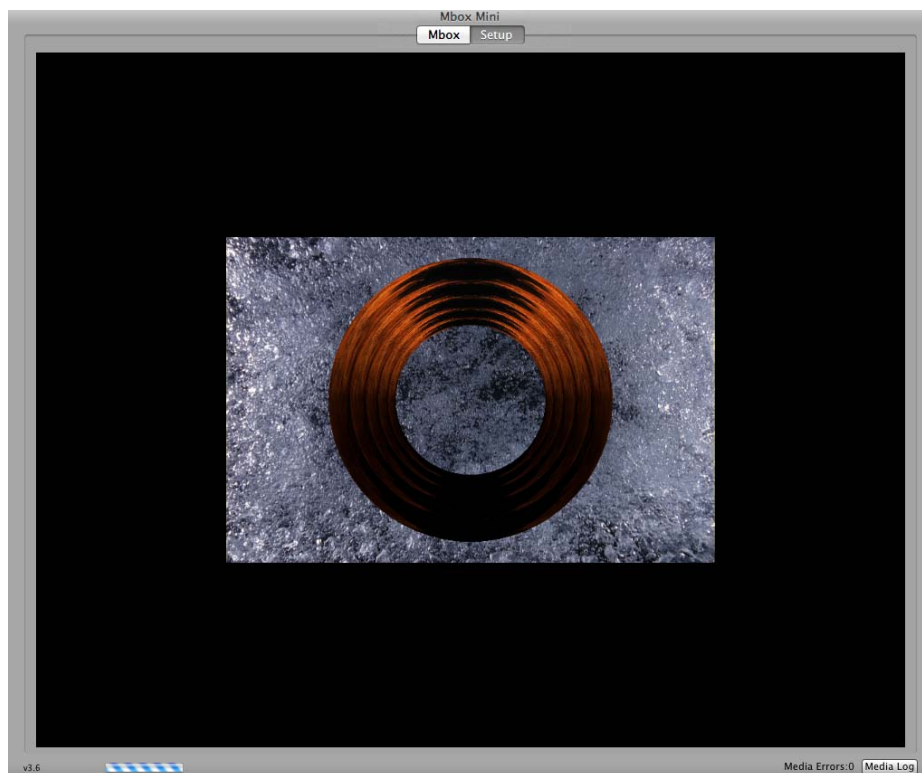
Upon launch, the **Mbox** tab will be selected. This tab shows the server's video output. The **Setup** tab is used to configure the server preferences and operating modes. Refer to "Application Window" on page 35 for more in-depth information about these functions.

While Mbox is running, there are two important key combinations to note:

+ To toggle between Fullscreen or Window mode, press [⌘ F].

+ To quit Mbox, press [⌘ Q].

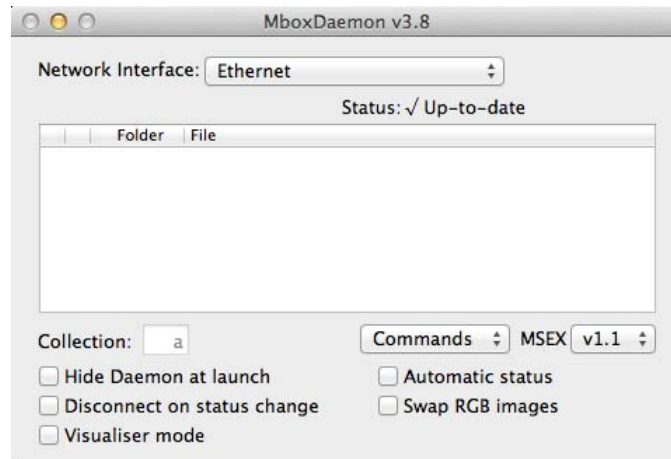Training Video: Mbox Media Server Training Video Chapter 4: Patching

Mbox Daemon

Mbox network communication involves a software application called Mbox Daemon. This software is required on all Mbox servers to enable remote feedback. When the Mbox application is launched (on the Mbox server), it will automatically launch the last version of Mbox Daemon used on that computer.

To view the Daemon interface, choose it in the dock or press [⌘ D] while the Mbox application is in Window mode.

The Mbox Daemon screen will always reflect the Network Interface setting as configured in the Setup tab (see above). The Network Interface setting can be changed at the Mbox Daemon screen, however, this is not recommended since the setting WILL NOT be retained if the Mbox is restarted. A sample Mbox Daemon screen is shown below:

Training Video: Mbox Media Server Training Video Chapter 7: Daemon Application

Adding Content

Use the following guidelines when adding content:

+ 2D content (Textures) must be placed inside the /Mbox/Media folder.

+ 3D content (Objects) must be placed inside the /Mbox/Models folder.

+ The use of both Textures and Objects relies on the correct structure of their parent folders and the correct numbering of folders and files. You may add up to 65535 Textures and up to 255 Objects.

+ /Mbox/Media Folder - Texture files may be numbered from 0 to 255 and must be placed into folders numbered 0 to 255 inside the Media folder. Folders 0 and 255 are used for built-in functionality and should be avoided, but can be used if necessary.

+ /Mbox/Models Folder - Object files may be numbered from 0 to 255 and must be placed into a folder numbered 0 inside the Models folder. The Mbox Mini installer will have created this folder and placed some files inside. You may replace these files as desired.

Refer to "Folder and File Numbering" on page 27 for examples of Media and Model folder and file structuring.

Patching Fixtures

In order to control Mbox, it will need to be "patched" to a control console. Patching allows channels to be mapped to the controllable elements of the Mbox environment. Use the following guidelines when patching:

+ The Mbox Mini software, using 4 layers of playback, requires at least 190 channels.
+ One control universe per server is required.
+ By default, Mbox Mini is configured to respond to Art-Net universe 0.

CAUTION! For ease of operation, the many channels that make up one Mbox server are divided into multiple fixtures. Unless you are an advanced user, it is strongly recommended that you patch all fixtures that are part of the Mbox profile. If you do not patch all of the associated fixtures, undesirable results will occur! The parameter mapping charts on the following page shows all fixtures that make up a single Mbox server. The fixtures must be patched in the order shown.

Parameter Mapping

Summary: Single Output

Start	Size	Type	Universe
1	11	Master	1st 190 Total Channels
12	8	Keystone	
20	14	Shutter	
34	39	Layer 1	
73	39	Layer 2	
112	39	Layer 3	
151	39	Layer 4	

Summary: Dual Output - Panoramic Wide

Start	Size	Type	Universe
1	11	Master	1st 190 Total Channels
12	8	Keystone	
20	14	Shutter	
34	39	Layer 1	
73	39	Layer 2	
112	39	Layer 3	
151	39	Layer 4	

Playing Back Video - Quick Start

Once a console has been connected, the Mbox fixtures patched, and the display device turned on, video can be played back.

IMPORTANT! The following procedure assumes that some content has been placed at Folder 001 / File 001.

To immediately play back video:

Step   1. At console, select Master fixture and set its intensity to full.

Step   2. Select Layer 1 fixture and set its opacity to full. (Note that when viewing in the HUD, opacity will be shown as "opac.") A white square should appear in the center of the display:
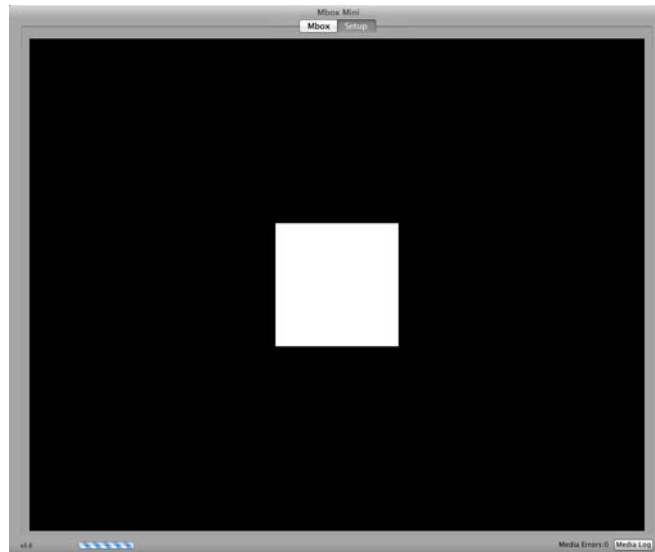
Step   3. With Layer 1 still selected, set Texture Folder to a value of 1. Then set Texture File to a value of 1. An image should appear on the display.

Step   4. At this point, adjust the folder and file parameters to change to different media content. For example, changing to another folder and file might result in the following clip:
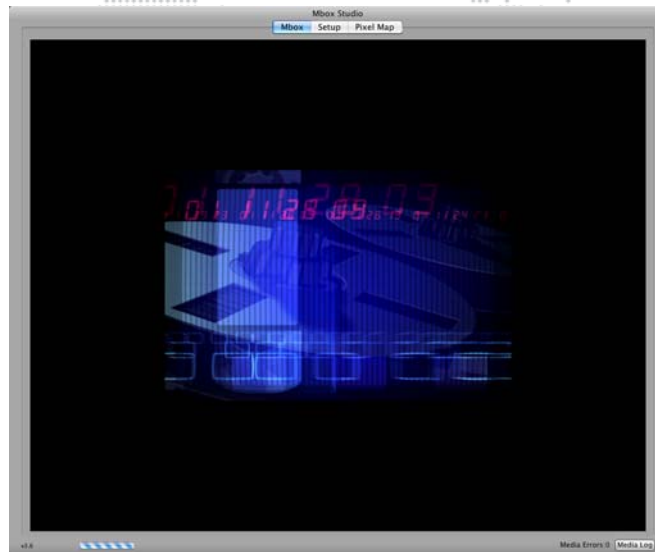
Step   5. To apply an effect to the layer, for example, set the Effect 1 parameter to 2 (sepia tone) and set the Effect 1 Control A to 255 (full). This should result in a sepia version of the same clip:
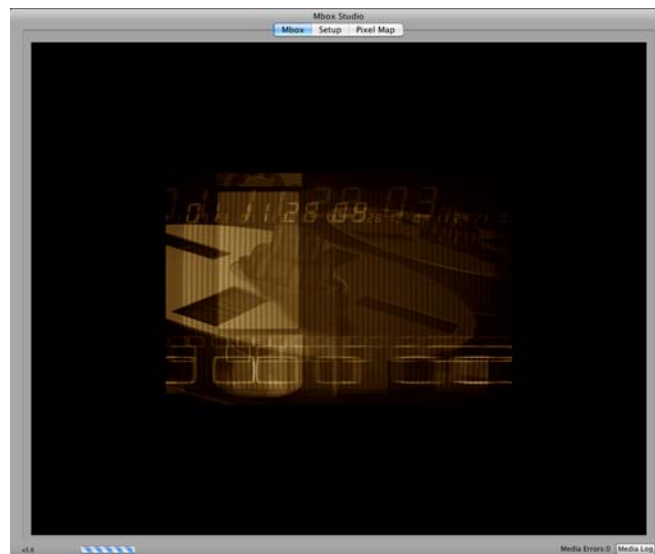
Notes

3.

CONFIGURATION

This chapter provides information regarding the initial configuration of the Mbox Mini software.

+ MEDIA CONTENT

+ APPLICATION WINDOW

+ HEADS UP DISPLAYS (HUDS)

+ ENHANCING MBOX PERFORMANCE

MEDIA CONTENT

File Type Specifications

Mbox supports Apple QuickTime® technology for rendering and playback of images and movie files. The following file types are supported:

Still Image Files

JPEG (.jpeg or .jpg), PNG (.png), TARGA (.tga), TIFF (.tiff or .tif) for still image files.

+ For still images not requiring transparency, JPEG is the best choice.

+ To use files with transparency, PNG is the best choice.

Movie Files

QuickTime format (.mov) for movie files. Spatial-compressed codecs will always provide better playback results than temporarily-compressed codecs (e.g. H264 and MP4 are not recommended and MPEG2 is not supported).

+ Photo JPEG – Medium (50-60%) is the recommended codec for movie clips.

+ Apple ProRes - Light or Normal are also recommended, High Quality can be used if you make sure the bitrate is not too high (150Mb/sec or lower).

Note: File extensions are not case-sensitive. For example, files with extension .JPG and .jpg are the same.

3D Model Files (Objects)

Custom 3D models can be added to the Mbox. The server supports the Alias Wavefront™ 3D .obj format. There are several third-party applications that can export files to this format, including Maya®, Poser®, Bryce®, and Cheetah3D to name a few.

File extensions for 3D object files:

+ .obj is the standard format.

Additional File Types

Scalable Vector Graphics (.svg) for digital gobo files, .aiff for audio files, and .rtf or .txt for text files.

Recommended Practices

JPEG and PNG file types are recommended for creating still image content for optimum output from Mbox. For movie content, Photo-JPEG (50% - 60% quality) is the recommended codec for optimum video playback. Mbox will play movies that use other QuickTime codecs, if the appropriate codec is installed on the server. However, sometimes those movies may not load as quickly or play as smoothly as movies that use the preferred codecs. For best results, all custom content should be created in a single, common format. The Photo-JPEG codec is strongly recommended. (Refer to "Creating Custom Content" on page 120 for more information.)

Note: In this manual, "preferred movie codecs" refers to Photo-JPEG, DV/DVCPro, Apple Intermediate, Animation, Apple ProRes 422 (Proxy, Light, Normal and High Quality), and Apple ProRes 4444. All other movie codecs are considered "non-preferred."

Typically, movie files should not have embedded audio tracks, although embedded audio tracks in movies can be made to play. To completely remove audio tracks from movies, use your preferred application to re-render or export the movie without its audio tracks.

If the file extension of a DV movie file is changed from .mov to .vid, Mbox will deinterlace the movie as it plays. The results of deinterlacing on the fly are not as good as when using deinterlaced content, but this means that interlaced content can be played without it looking as noticeable.

As is the case with all media servers, Mbox playback performance can decrease as the resolution of its media files increases. Content up to and including 1080p can be played, keeping in mind that higher-resolution files will sometimes lead to lower performance than smaller files. The amount of compression used when creating a file will also affect performance. Given equal file resolutions, higher quality (lower compression) files may not play as well as higher compression (lower quality) files. Depending on the final display device, high-resolution or low-compression files may not look better than lower-resolution, more highly-compressed files. For best results, try to find a compromise between quality and performance.

Media Content Management

Mbox Mini allows for a flexible media folder structure. Media can reside locally on the Mac hard-drive or on additional hard-drives or storage devices connected to the computer. Just be aware that using external devices is not recommended because playback performance can suffer.

At startup, the Mbox Mini software performs a media scan to search all folders that reside within the *Mbox/Media* and *Mbox/Models* folders on the hard-drive of the Mbox server. External media can be attached to these folders using aliases, provided the following guidelines are observed:

+ Aliases must link to folders on the supplemental media; they cannot be aliased to files.
+ An alias cannot refer to another alias.
+ The */Mbox/Media*, and */Mbox/Models* folders cannot be aliases.

Each time the Mbox server completes a media scan, a text file is created with all media files found and their corresponding combined file numbers. This text file can be found at */Mbox/Media_Report.txt*

Note: The initial media scan allows the locations of files to be resolved at startup, therefore the "distance" of a file from the main folder has no effect on the time required to find a file. In addition, the content in the *Media* folder is allowed to be nested (folders within folders) to a limit of 64 folders deep.

Folder and File Numbering

Media content files are arranged into individual "Library" folders within the */Mbox/Media* or */Mbox/Models* folders. Each folder name must begin with a 3-digit index value. In turn, each file within the Library folder must be named using a 3-digit index value between 000 to 255 as a prefix in the case of the Media folder and must start with an index of 000 in the case of the Models folder.

Control channels on the console will be used to select a Library folder and the numbered file within that folder.


```
📁  Mbox/Media
     └──▶ 📁  001.videos
               ├──▶ 001.rain.mov
               ├──▶ 002.drops.jpg
               └──▶ etc.
```
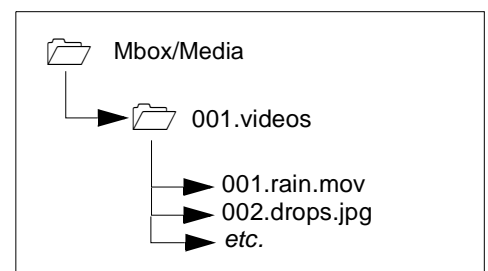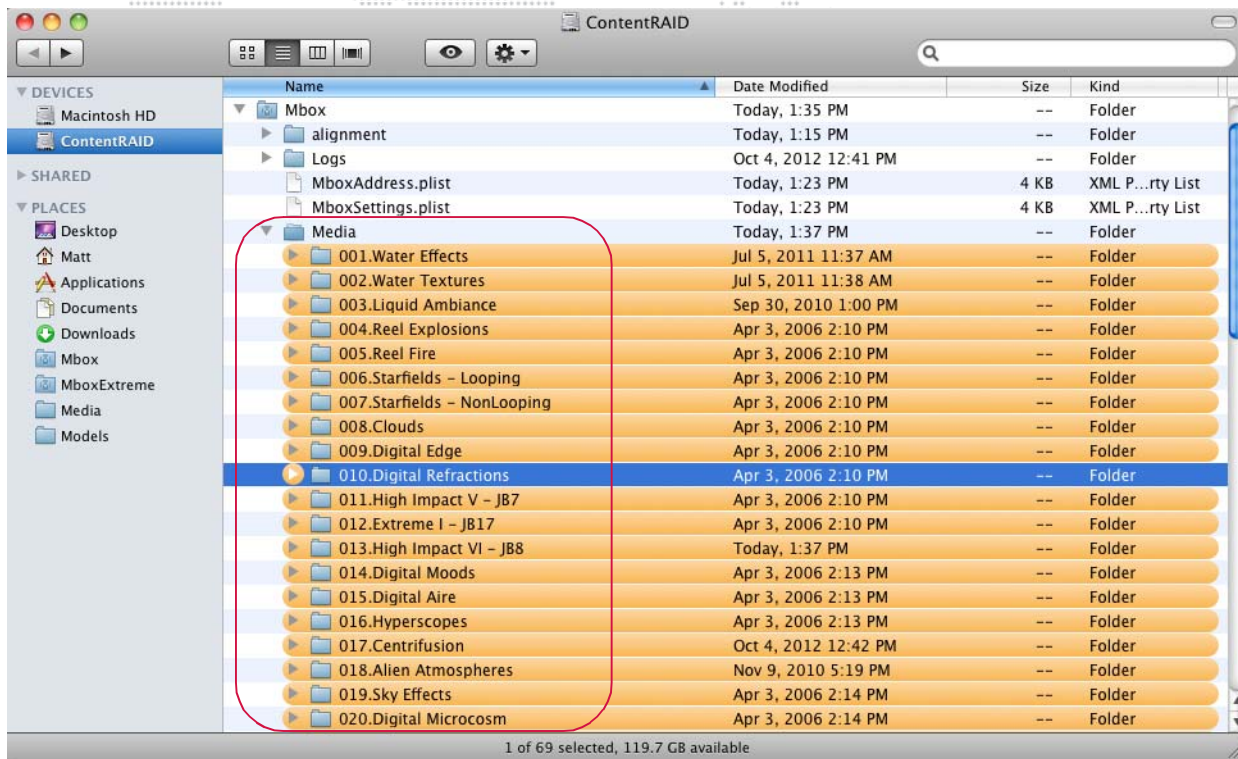
Library folders within the *Media* folder must have a number prefix and can have a descriptive name separated from the number by a period or a space. For example:

025.Central Park or **025 Central Park**

Files within each library folder must have a numerical prefix, can have a descriptive name if desired, and must have a three-letter file extension. Each of these parts should be separated by a period:

001.Great Lawn.jpg or **001.jpg**

Movie and still image files may be numbered between 000 and 255 and placed in folders numbered 001 - 254. Models and digital gobos may be numbered between 000 and 255 and placed in folders numbered 001 - 255.

Avoid using folders 000 and 255 in the *Media* folder. You must use folder 000 in the *Models* folder.

Stock 3D objects, animated gobos, and digital gobo files are located in *Models* folder 000.

Additional Media Content Organization Guidelines

+ Media folders 000 and 255 are reserved for default files and utility files respectively. The Mbox Mini software can use a video capture card to receive external video input. This input can be used to display video from an external source on any layer, as if it were content on the server's hard-drive. You may also display video from a USB or FireWire® camera connected directly to the computer. To view video from the video inputs, select library 255 and either texture 254 or 255. The Setup tab allows assignment of up to two video inputs to these values. If no video capture card or camera is detected and diagnostics are enabled, it will show the "No Camera" image on the display when requesting one of these media locations.

No Camera

+ When adding content files, DO NOT duplicate existing folder and file number combinations. It is best to make new folders for custom content.

+ Leading zeros are not required when numbering folders or files, but they can help by making the file listing easier to read.

+ Folder 255, texture 000 will display the Mbox Art-Net universe and start address.

+ Mbox Mini scans for content upon startup. If new content is added while Mbox Mini is running, the Media and Models folders must be re-scanned in order to use that new content. These folders can be re-scanned while Mbox Mini is running by pressing [⌘ R] on the server's keyboard or from the console by using the control channel on the Master fixture. There will be a brief pause in movie playback while the rescan takes place.

Address

+ Media Folder 255 contains the following:

> 0: Mbox patch info
>
> 121-124: reserved for Layer Copy FX+ functionality for Layers 1-4 respectively
>
> 201-202: reserved for Syphon inputs
>
> 254-255: reserved for video inputs
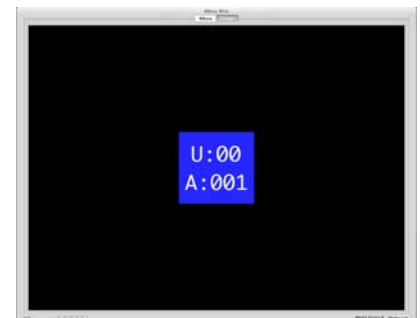
Note: Refer to "Layer Copy" on page 68 for more information about the Layer Copy feature.

Adding Media Content

When adding content files, it is recommended that new, unique folder numbers be created for the custom content. It is possible to have more than one folder with the same numerical prefix, however, as long as the content files within such folders do not have the same numerical prefix. Also, folders need not have numerical prefixes.

CAUTION! Numbered content files will be assigned a folder number that is equal to the numerical prefix of their immediate parent folder. Therefore, using folders without numbers requires caution.

A file at this location: *Media/022.Water/BlueWater/001.SomeWater.mov* - will be assumed to be in folder 000, because the immediate parent of the numbered file has no numerical prefix

The correct method: *Media/BlueWater/022.Water/001.SomeWater.mov* - would work as desired, assigning the file to folder 022.

To add movies or image files:

Step   1.   On the content drive, open *Mbox/Media* folder.

Step   2.   At **File** menu, select **New Folder** and enter a unique 3-digit number (that is not already being used) followed by a descriptive name. For example, "075.mycontentfolder".
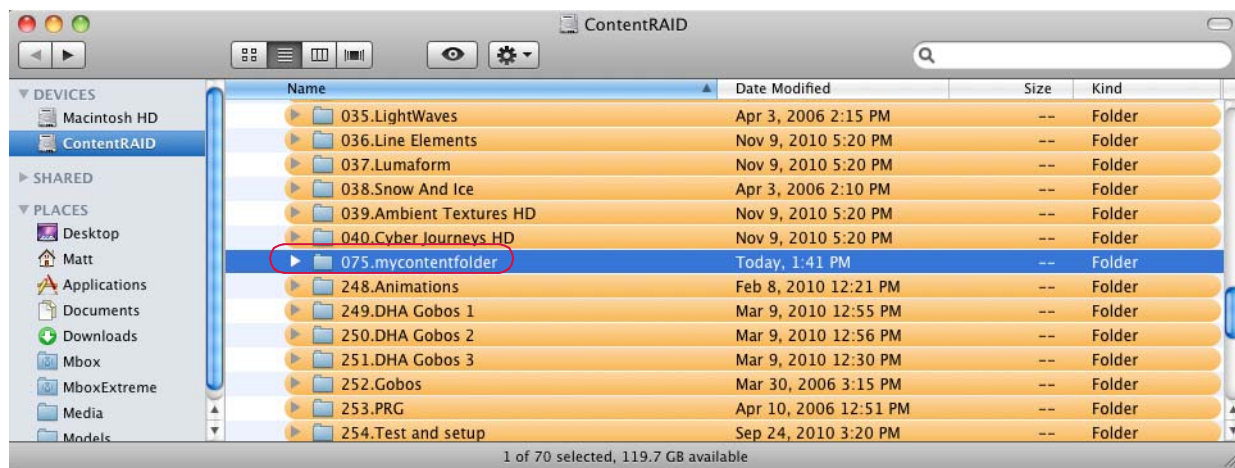
Step   3.   Copy image and/or movie files into the new folder.

Step   4.   For each file, add a unique 3-digit index value to each file (between 0-255) along with a descriptive name that includes the file extension. For example, "001.earth.jpg", "002.wind.mov", "003.fire.mov", etc. (Note the caution given above.)

To add 3D objects:

Step   1.   On the content drive, open *Mbox/Models* folder.

Step   2.   Place files into the 000.Objects folder.

Step   3.   For each file, add a unique 3-digit index value to each file (between 0-255) along with a descriptive name that includes the file extension. For example, "001.man.obj", "002.hammer.obj", "003.airplane.obj", etc. (Note the caution given above.)

Grouped Content

Description

The Mbox applications support the use of "content grouping" in order to allow servers to have content for all servers stored on their hard-drive, but to allow each server to play only the content desired for its own output(s). In this manner, content synchronization can ensure that all show content is on all servers (allowing for one server to take over for another). When programming, selecting a folder and file will cause content (with the matching folder.file index) to be played from within the server's corresponding group folder only.

To facilitate keystone, warping, and projection mapping setups, each output of each server can be given a numerical "Identity" value (between 0 and 255). The Identity property is also used for content group designation. In the content database indexing system for Mbox, the content group is a prefixed 8-bit value - i.e. Group.Folder.File, which each element being a number 0 - 255. (This concept essentially extends Mbox's content indexing system to be 24-bit rather than 16-bit as prior to v3.6 software.)

If content is not assigned a group number, then it will always be played when a particular Folder and File are selected. Standard rules for file conflicts (repeated Folder.File indices are conflicts) will apply. If content is assigned a group number, then there may be multiple Folder.File indices with the same values. But since each will have a unique Group number, there are no conflicts. The playback of content with the matching group index is automatic and no parameter on the controller (console or Mbox Director) selects this value. Users will preset the Identity of each output of each server in order to dictate what content is played.

Usage

The typical usage for grouped content will be when more than one Mbox server is used as a portion of a multi-server (widescreen, matrix, etc.) setup with coordinated outputs. Often these setups will make one larger image with each server supplying one or two sections of the larger image. The content grouping scheme allows the programmer to select the same Folder and File values for all servers and get the appropriate portion of the larger image to play on each server.

Note: Content Grouping does not chop up or distribute content automatically to the relevant server. It merely causes the server's designated group's content to play when a folder.file is selected.

Usage is not limited to multiple servers making up one larger coordinated raster. Grouped content can be used as a tool for content organization even when disparate content is used on multiple servers.
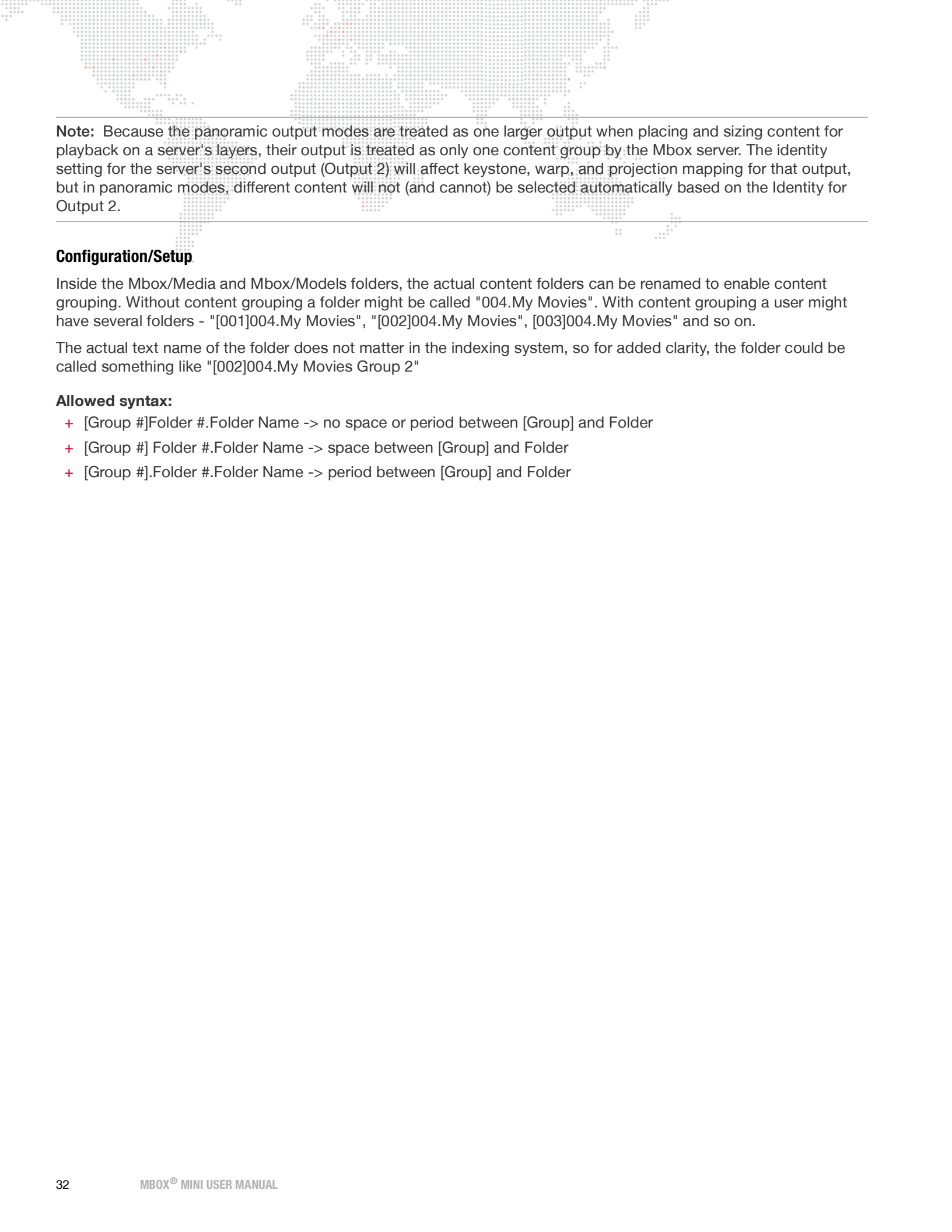
Example:

Imagine a system with three servers, each having two outputs in Dual Independent mode. This system has six total outputs, and for a wide blend, each output will display 1/6th of the total image. For best playback results, the user may choose to supply six unique pieces of content each being approximately 1/6th of the final image (plus overlap as required).

Prior to Mbox v3.6 the user could place all six pieces of content into the same folder and give each piece a unique file number. (e.g. 001.001, 001.002, 001.003 and so on…) When programming the show, the user would have to select six different pieces of media for playback, a different file on each server.

Now, the user would first designate the Identity number for each output in the system (e.g. 1, 2, 3, 4, 5, and 6). Then the user would create six different content folders (one for each group identity) on each of the three servers, labeling them uniquely (see below for details) with numbers to match Group 1, Group 2, Group 3, etc. but each having the same folder index number. Inside each folder the user would place the content (1/6th of the final content image) that matches that output's position in the overall scheme. The individual content files that make up one larger image would all be given the same file index number.

With the identity set in this way the user could then select folder 001 and file 003 for playback on all the servers. The output designated as Identity 1 would play Group001.Folder001.File003, and so on…

Note: Because the panoramic output modes are treated as one larger output when placing and sizing content for playback on a server's layers, their output is treated as only one content group by the Mbox server. The identity setting for the server's second output (Output 2) will affect keystone, warp, and projection mapping for that output, but in panoramic modes, different content will not (and cannot) be selected automatically based on the Identity for Output 2.

Configuration/Setup

Inside the Mbox/Media and Mbox/Models folders, the actual content folders can be renamed to enable content grouping. Without content grouping a folder might be called "004.My Movies". With content grouping a user might have several folders - "[001]004.My Movies", "[002]004.My Movies", [003]004.My Movies" and so on.

The actual text name of the folder does not matter in the indexing system, so for added clarity, the folder could be called something like "[002]004.My Movies Group 2"

Allowed syntax:

+ [Group #]Folder #.Folder Name -> no space or period between [Group] and Folder
+ [Group #] Folder #.Folder Name -> space between [Group] and Folder
+ [Group #].Folder #.Folder Name -> period between [Group] and Folder

Creating and Viewing Content Thumbnails

The Mbox Mini software can create thumbnail images of content placed on the hard-drive. To force an update of thumbnails by Mbox Mini, press [⌘ T].

Thumbnail images are stored in a standard location on every server and are grouped with HTML files that allow any networked computer (Mac or PC) to use an HTML browser to view the thumbnails. Individual thumbnails can also be used by consoles connected to the server. The thumbnail files are located in folders on each server at */Library/ Application Support/Mbox*. The HTML pages can be browsed by entering *http://IP Address of remote server* in the address line of the browser.

CAUTION! While thumbnails are being created, playback performance on the server may be severely affected. Thumbnail creation time will depend on the amount of content in the *Media* and *Models* folders. While thumbnails are being created, you will not be able to view them from either a browser or the Mbox Remote application.

Console Interface

Once patched to an automated lighting console, Mbox can be controlled using the console's interface. The following example shows the Media window offered by the PRG Vx76 series console. In this case, the window is displaying thumbnails for the Texture library media files. Basic controls - such Play, Stop, Rewind, Loop, etc. - are also available in the window.
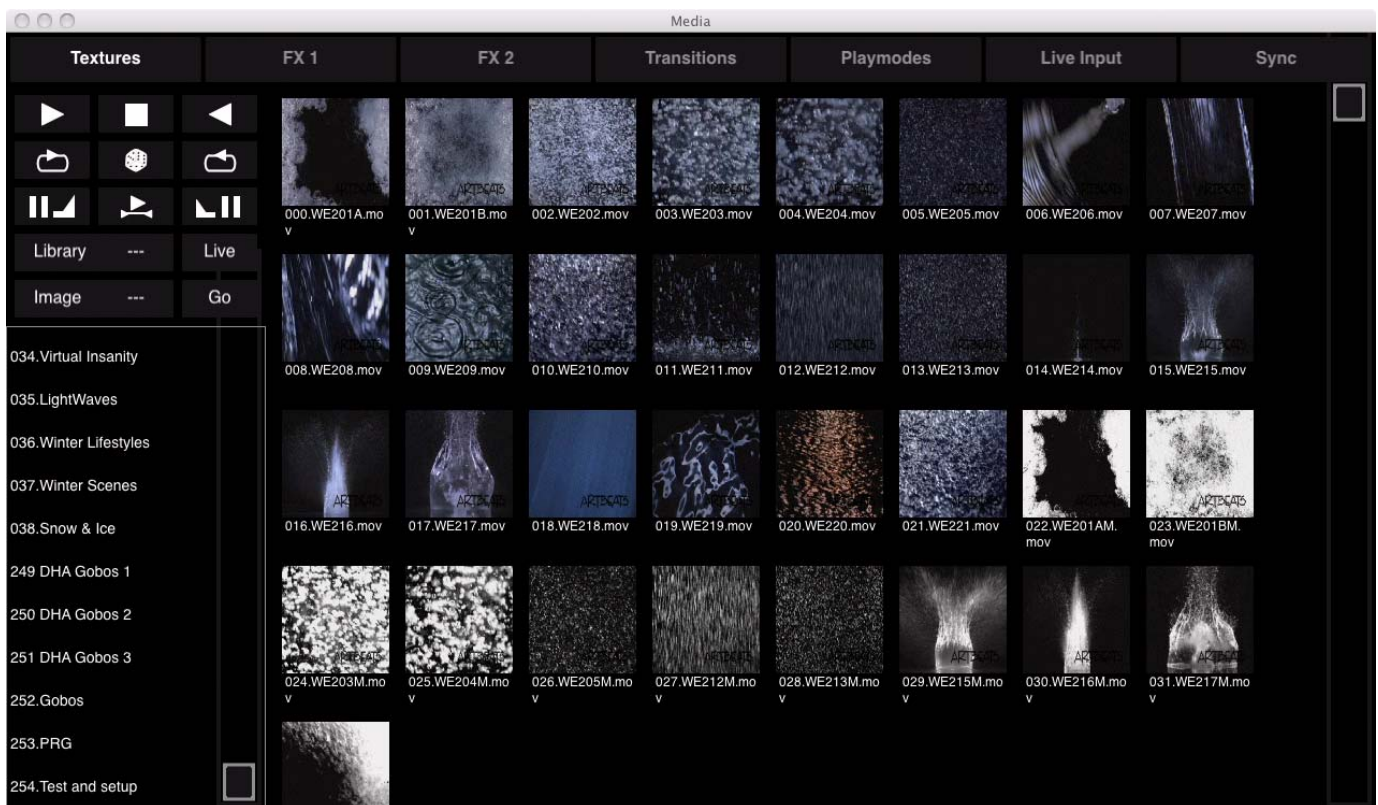
▶ Training Video: Mbox Media Server Training Video Chapter 6: CITP – Streaming Feedback & Thumbnails

▶ Training Video: Mbox Media Server Training Video Chapter 20: Codes, Storage, and Media Types

Viewing Thumbnails in a Browser

Another way to view media thumbnails on an Mbox server without a CITP-connected console or Mbox Director, is to use the web browser of any computer connected to the Mbox network. Enter the following into the browser:

+ If using a browser on the local Mbox sever, the address **http://127.0.0.1/Mbox** can be used to view the local thumbnails.

+ To view the thumbnails on a remote machine, enter that machine's IP address. For example: **http://192.168.0.123/Mbox**

When accessing the web pages from a rental Mbox server, the **/Mbox** portion of the address can be omitted.
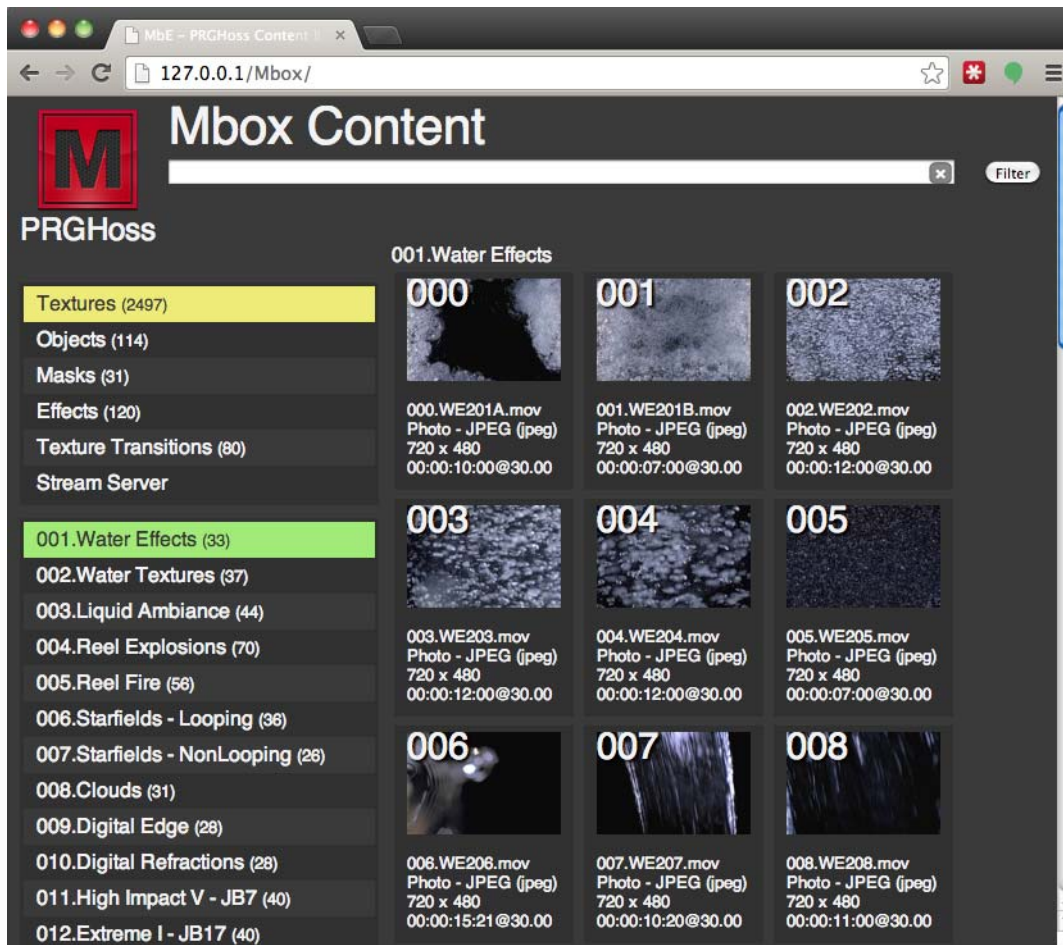
APPLICATION WINDOW

Overview

The Mbox Mini application window allows the server's output to be viewed in either Window or Fullscreen mode on the preview monitor.

Fullscreen - While in Fullscreen mode, the server's output (i.e., video) will be displayed at "full screen" without any user interface showing.

Window - While in Window mode, the application's user interface will be displayed. The Window mode interface contains two tabs, Mbox and Setup, as shown in the sample screen below:
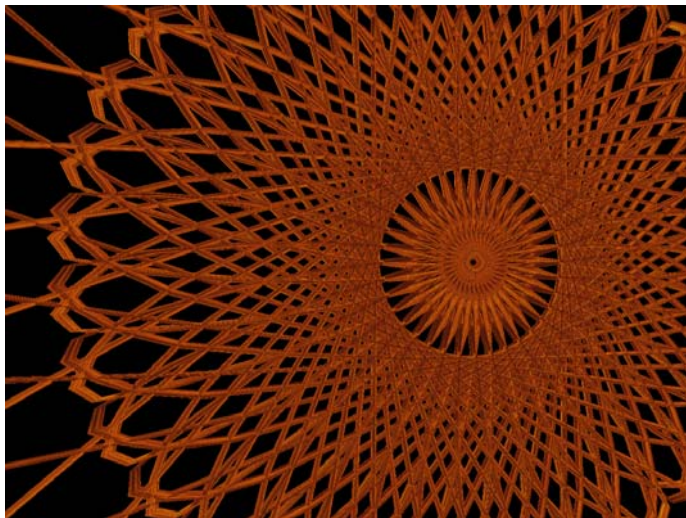
Fullscreen Mode	Window Mode
Output to: Preview (Monitor) and Stage	Output to: Preview (Monitor) only
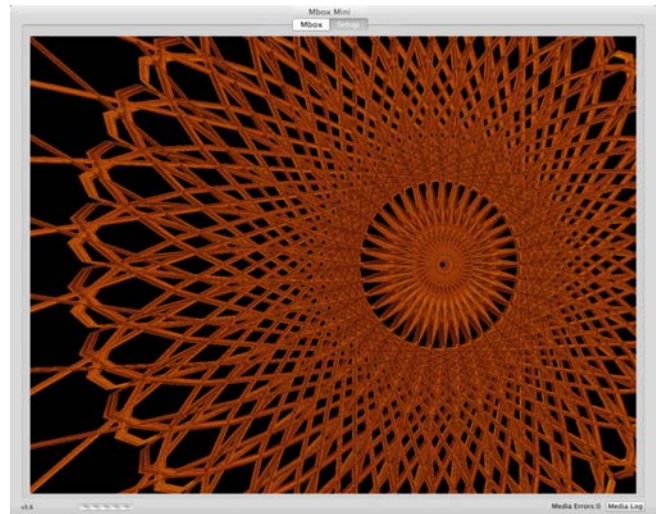
The function of the two Window mode tabs is as follows:

+ **Mbox tab** - used for previewing the server's video output.
+ **Setup tab** - used to configure the server. Refer to "Setup" on page 36 for more details.

Setup

The Setup tab of the Mbox application window is used to:

+ Set the operating mode and Art-Net universe for the Mbox server.

+ Set the application to start in either Window or Fullscreen mode. For Window mode, the window size can be set. For Fullscreen mode, the resolution can be set.

+ Input License Key to authorize the application.

+ Set a width and height for Textures.

+ Set the network port for CITP communications between Mbox and Mbox Remote and/or any lighting console or visualizer that supports CITP and MSEX. (Refer to the Mbox Remote User manual for more information.)

+ Configure video inputs.

+ Verify the configuration of the connected I/O module output sections.

+ Enable or disable Diagnostics, Deinterlace, Pixel Mapping, and Audio Playback.

Sets Window Mode Preferences   Sets Fullscreen Mode Preferences

Sets Active Area Preferences

Sets operating mode, # of layers, protocol, universe, and source IP for the Mbox server

Sets Screen Arrangement Preferences

Sets Local Video Inputs

Sets Sync Network Port

Sets Management/CITP Port

Licensing Actions and Display

Opens Media Log

"Diamonds" indicate that if this setting is changed, a software restart is required for the changes to take effect.

Sets Texture Size

Sets I/O Module Output Configuration

Sets Miscellaneous Options

Rendering Rate (Update Rate Divisor Setting)

Refer to the following pages for in-depth explanations about each of these settings.

Note: The settings configured at this window are stored in a standard Mac Property List (.plist) format. This file, located at */Mbox/MboxSettings.plist*, stores the startup settings for the Mbox software and the variables that define its operating parameters: window size, screen size, etc. The most important of these settings can be adjusted from the Setup tab, however, some advanced settings can only be adjusted using Mbox Remote. (Refer to the Mbox Remote User manual for more information.)

Setup: Control

The Control section is used to set the operating mode, protocol, address and layer count for the server.

+ **Mode** - use this pop-up menu to select the desired mode. (Refer to "Operating Modes" on page 14 for detailed information about each mode.) It is important that the operating mode match the console profile that is being used. If these are not properly matched, there may be little or no control over the Mbox server!

+ **Active Layers** - use this pop-up menu to select the number of active layers for the server.

+ **Protocol** - use this pop-up menu to specify whether Mbox will receive Art-Net or sACN (Streaming ACN). This selection applies to all the control protocol that Mbox receives (i.e., it all has to be of this type) whether it's for normal control, or for the TC_DMX, Pixel Map Group controls, Pro Lighting, etc. Art-Net allows universes 0-255 and sACN allows universes 1-63999, both having 512 addresses (channels) per universe.

+ **Universe** - this field refers to the universe that Mbox is "listening to" for control information. Mbox responds to Art-Net data even when DMX512 is connected to the Dual I/O module. The I/O module incorporates a built-in DMX512-to-Art-Net converter whose two DMX ports are pre-set to universe 0 and universe 1. Therefore, the server's universe should be set to 0 for for use with DMX512 control. If using a direct Art-Net or sACN connection, then modify the server's universe to match that of the control signal. Mbox numbers control universes using decimal notation: 0 through 255 for Art-Net, and 1 through 63,999 for sACN. Other devices may use different notation for universe numbers so some conversion between numbering systems may be required.

+ **Source IP** - this field shows the IP address of the device that Mbox is currently receiving control data from on the universe that is set in the Control Universe field. The IP address is shown to help indicate that Mbox is receiving data on the selected universe, and as a reference for troubleshooting.
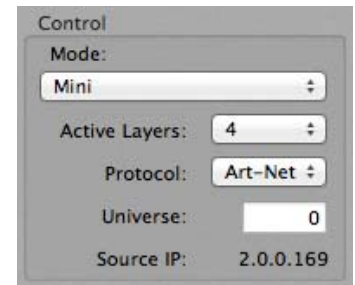
Setup: Window Mode

The Window Mode section is used to configure the behavior of the Mbox application window when not running in Fullscreen mode. This is intended to be an accurate representation of the fullscreen output while in Window mode.

+ **Scale %** - use this pop-up menu to specify a window size in relation to the fullscreen size. This number is a percentage of the Fullscreen mode. For example, choosing 75 would mean that the Window mode size is 75 percent smaller than the Fullscreen size. The resulting window size will be shown under the menu (960 x 540 in the example to the right).

+ **At Startup** - when this box is checked, the Mbox application will start in Window mode. If it is not checked, the application will start in Fullscreen mode.

+ **Aspect Comp.** - when this box is checked, Mbox will change the aspect ratio of Window mode to compensate for the aspect ratio value selected in the Fullscreen Mode setup area. By default, the Window mode will be sized to an exact percentage of the Fullscreen mode dimensions (50% of 1920 x 1080 is 960 x 540) and the Fullscreen aspect ratio is ignored. However, to have the aspect ratio of the Fullscreen mode taken into account in Window mode, then check the "Aspect Comp" checkbox. This will cause the Window mode dimensions to take the Fullscreen aspect into account, sizing the window appropriately. Maintaining the correct aspect is critical for proper aspect viewing in Window mode. *For example*: NTSC SDI has a resolution of 720 x 480, but a true aspect ratio of 1.364. A window scale setting of 50% results in a window size of 360 x 240, which is an actual aspect ratio of 1.5 meaning that the image will be stretched. Checking the "Aspect Comp" checkbox will result in a window that is 327 x 240, thus keeping the aspect ratio at 1.364.
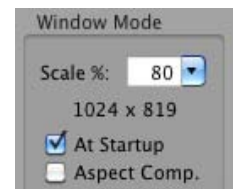
Setup: Fullscreen Mode

The Fullscreen Mode section is used to set the resolution, frequency, and aspect ratio of the rendering context that Mbox uses. It is not uncommon to change the Fullscreen Mode settings since, for example, a standard definition projector would require different settings than an HD projector, etc.

Getting pixel accurate results from any Fullscreen mode setting requires the computer to be receiving EDID information that matches the Fullscreen mode settings. If no EDID is present or if the EDID does not match the settings entered here, Mbox will not output the desired resolution and/or frequency when the software enters Fullscreen mode. Always confirm that you have the appropriate EDID before using the Mbox software. When you are using Mbox without the I/O Module, you will need to supply the appropriate EDID by connecting the output device (or some alternate EDID spoofing device) to the computer's video output(s).

If "For Current Display" is selected, then the resolution, frequency, and aspect values will be automatically set to match the settings currently being used for the corresponding output of the computer (i.e., the current Mac desktop settings). If one of the standard modes is selected (e.g. SDI 720p), the resolution, frequency, and aspect values will be automatically set to the correct values for that mode. If the custom mode is selected, then resolution, frequency, and aspect will need to be manually adjusted. For additional sizing and positioning options, the selected FullScreen Mode setting can be combined with the "Use Current Mode" checkbox and the AutoFit pop-up (see next page). If the custom mode is selected, the Active Area Controls become available for use.

The Fullscreen mode settings will also be applied to the video signal that is output from the computer when Mbox goes into fullscreen mode. Ensure that the preview monitor is able to display an image when receiving a signal with that resolution and frequency. If the native aspect ratio of the monitor does not match that of the Fullscreen mode, then it may still display an image, however, it will have the wrong aspect ratio.

Above all, it is most important to tailor the Fullscreen mode settings to the stage device, than to the preview monitor. Custom EDID management systems may allow output resolutions to be set that may not work with some preview monitors.

When a certain resolution setting for Fullscreen mode is requested, the operating system will review its EDID information to see if it is capable of that configuration. If a resolution is requested that does not match a resolution listed in the current EDID set (see "Connected Display Devices and EDID" on page 46 for further details) then the operating system will pick a resolution from among those in the current EDID. In this case, Mbox has no control over what resolution the operating system provides. The selected resolution may be larger or smaller than the desired Fullscreen mode size. If this happens, there are several possible outcomes which are controlled by the selection made in the AutoFit pop-up (see next page).

CAUTION! It is strongly recommended that the Mac desktop resolution and refresh rate (frequency) be set to the same value as the Fullscreen mode settings. If these settings are not the same, then any devices connected to the media server's outputs (both Preview and Stage) will temporarily lose sync when switching between the Fullscreen and Window modes or when the Mbox software is relaunched. With some display devices, the results of losing sync, even temporarily, can be quite disruptive. With identical settings for the Mac desktop and Mbox Fullscreen mode, there will be no loss of sync. (Desktop resolution can be set in the Mac's Preferences window.)
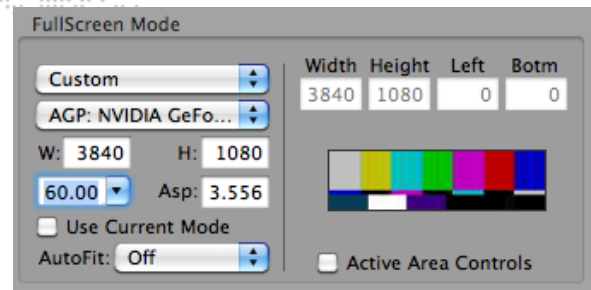
+ **Use Current Mode** - This checkbox is used to force Mbox to always use the current desktop resolution and frequency, no matter what Fullscreen Mode resolution and frequency are selected by the controls above. The fullscreen context will be positioned and scaled to fit that resolution based on the selections made in the Fullscreen Mode and AutoFit pop-ups.
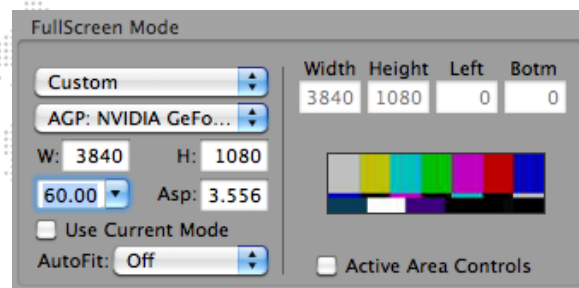
+ **AutoFit** - The Autofit pop-up provides several options:
  - **Off** - If "Off" is selected, the Fullscreen Mode context will be placed into the bottom-left corner of the received resolution. If the Fullscreen Mode is smaller than the received resolution, then Mbox will show images with a black border around them. If the Fullscreen Mode context is larger than the received resolution, then a portion of Mbox's images may extend off the screen to the right and top.
  - **Underscan** - If "Underscan" is selected, the Fullscreen Mode will be centered in the received resolution and automatically scaled up or down proportionally until both the width and height fit into the received resolution. The entire Fullscreen Mode context will be visible. Aspect ratio will be maintained.
  - **Overscan** - If "Overscan" is selected, the Fullscreen Mode will be centered in the received resolution and automatically scaled up or down proportionally until either the width or height fits into the received resolution. Portions of the Fullscreen Mode context will not be visible. Aspect ratio will be maintained.
  - **Anamorphic** - If "Anamorphic" is selected, the Fullscreen Mode will be centered in the received resolution and automatically scaled up or down until both the width and height exactly fits into the received resolution. Aspect ratio will not be maintained if the aspect of the context does not match the aspect of the received resolution.

+ **Active Area Controls** - In some cases, it may be necessary to use a specific output resolution but keep content fixed inside a smaller area of that resolution. For this purpose, Mbox provides a way to embed a custom active screen area within the larger output area. To use this feature, select "Custom" from the pop-up menu, then check the "Active Area Controls" checkbox. Enter values for Width and Height to define the smaller display area. The values for Left and Bottom can be used to position the interior area at a specific location within the larger area. Note that Active Areas may not work as intended if the "Use Current Mode" checkbox is checked and if the AutoFit pop-up has a selection other than "Off."

Note: Remember that Mbox screen origin is the bottom-left of the output raster. Also, when Active Area controls are enabled, the Window mode size will be a percentage of the Active Area size rather than the Fullscreen mode size.

Setup: Rendering

Mbox will always try to output video frames to the Mac graphics card at the same rate as the current output frequency. However, in some circumstances (e.g., a heavy load), Mbox is unable to deliver the full number of frames per second, and must automatically reduce the number of frames it delivers to the graphics card.

At the default setting of x1, Mbox will automatically adjust its update rate. However, if there is poor playback smoothness, it may be necessary to force Mbox to output a lower number of frames per second to the graphics card. The optional setting of ÷2 will force Mbox to output a number of frames per second equal to the numerical value that is 50% of the current fullscreen frequency (e.g., If fullscreen output frequency is 60Hz, then update rate would be set to 30fps.)
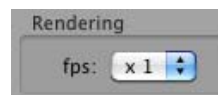
Note: This setting does not affect the output frequency from the graphics card.

Setup: License

The License section is used to display or enter a license key, as required.

If the software is unlicensed, then the video output will be watermarked, Syphon output framerate will be reduced, and all pixel mapping Art-Net output will be disabled.
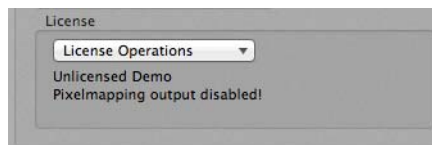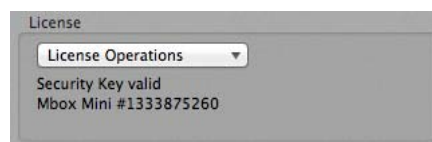
Setup: Texture Size

The Texture Size section is used to set the resolution of the largest content (still or movie) that will be displayed.

This setting should be kept to the lowest possible minimum, since overly large values can negatively affect performance. If you attempt to display content that is larger than the Texture Size - either in width or height - it will automatically be reduced in size (in 50% increments) until it fits within the Texture Size. This automatic resizing affects image quality, but it is necessary to ensure oversized content can be displayed quickly. We recommend finding a compromise between keeping the Texture Size as small as possible and keeping your image quality as high as required.
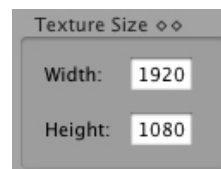
Because the majority of stock content on the Mbox hard-drive is 720px x 480px, the Texture Size should never be set lower than this value if you intend to use any stock content.

CAUTION! If changes are made to the Texture Size, the software must be restarted for these changes to take effect.

Setup: Screen Selection (Screen Arrangement)

When using the Single Output operating mode, Mbox can be setup to enter Fullscreen mode on Output 1 or Output 2 of the computer. The other output will continue to display the Mac desktop. Alternately, the "Mirror Screens" checkbox can be selected. This will cause the single output to appear on both screens at the same time.

When using either of the Panoramic Wide or Panoramic Dual operating modes, it is possible to set and adjust the overlap between the two screens and specify whether they are horizontally or vertically arrayed. Refer to for more information.
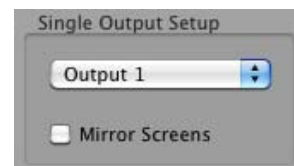
CAUTION! Mirror mode should not be used in show situations because it will cause one or the other of the output screens to tear. The functionality is included as a convenience. If you require a duplicate of either video output, you should use the appropriate video signal D/A or splitter.

Setup: Network

The Network section is used to configure the Sync, Management, and CITP settings when connecting two or more Mbox servers.

+ **Sync** - sets which port the server will use for layer-to-layer and network timecode sync.

+ **Management/CITP** - selects the port that the Mbox Daemon application uses for communication with consoles and the Mbox Remote application. When changes are made here, it will be reflected 4-5 seconds later in the Daemon screen. (Refer to the Mbox Remote User manual for more information.)
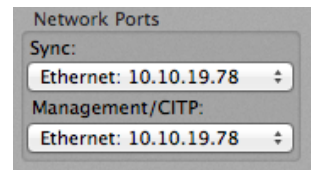
The Sync and Management/CITP pop-ups will populate to display the the name and IP address of the connected and configured interfaces. When the interface is not connected, the text will display in a gray color with no IP address to indicate that the once-configured interface is not currently connected.

The available port choices will depend on the computer and which Ethernet cables are connected on the servers. It is important to plug in network cables first before making changes to these settings. The settings will be remembered in case the cable is ever unplugged, but if the cable is disconnected the pop-up may not show the port name because it is not active.

+ "Local" is the default setting, but it only allows communication internally. To allow communication to another computer, an active Ethernet port must be selected.

+ "Ethernet 1" is typically reserved for use with Art-Net input to the Mbox computer, but may be used for Management/CITP in some cases (i.e., when using a controller that only uses one network port for Art-Net and CITP, such as Mbox Director).

+ "Ethernet 2" (or additional ports) is recommended for all other communications besides Art-Net, if at all possible. This is done to keep non-critical traffic from interfering with the commands sent to Mbox from the controller.

▶ Training Video: Mbox Media Server Training Video Chapter 6: CITP – Streaming Feedback & Thumbnails

Setup: Video Inputs

The Video Inputs section is used to select and configure the local video inputs to the server. These inputs can be a Video Capture card (e.g., Blackmagic Decklink), Thunderbolt input (Decklink UltraStudio), or an optional USB or FireWire camera.

The default setting is a particular resolution, so this may need to be changed to work with all resolutions that are being received by the Mbox server.

When setting up the video inputs, make sure that none of the layers are attempting to display the video input that is being configured. It may also be necessary to open the Decklink Preference window (in the Mac System Preferences) to select the type of signal that is being received (SDI, Composite, or Component) by a Decklink video capture card.

Refer to "Video Input" on page 107 for more information.
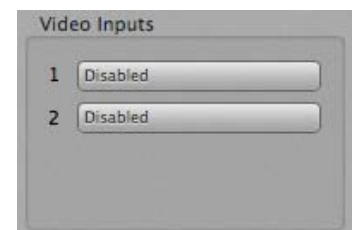
Setup: I/O Module

Note: Mbox Mini supports the use of the Mbox Extreme Dual I/O Module, but does not come with an I/O Module. This section would only apply if the Mbox Mini computer were connected to a Dual IO Module obtained from PRG.

The I/O Module section is used to verify the configuration of the connected I/O module output sections. This is important when using Dual Independent mode, as Mbox must know which output to dim with Camera 1 and which output to dim with Camera 2. When using Dual Independent mode, if Camera 1 dims Output 2 and Camera 2 dims Output 1, then the output sections are reversed.
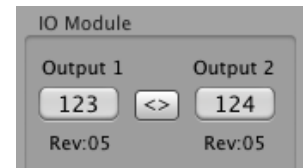
The buttons function as follows:

+ **Output 1 / Output 2** - The numbered buttons underneath Output 1 and Output 2 indicate that an I/O module output section has been found. If "---" is displayed on one or both buttons, this indicates that Mbox cannot find the I/O module circuit board in question.* These buttons also identify which output section is which. When the **Output 1** button is pressed, the top I/O module output section should flash its LEDs rapidly. Likewise, pressing the **Output 2** button should cause the bottom output section's LEDs to flash rapidly.

+ **< > Button** - If the **Output 1** button flashes the bottom LEDs and the **Output 2** button flashes the top LEDs, press the "**< >**" button to swap the order. Verify the swap by pressing the numbered buttons again.

    * If this occurs, check all USB connections between the Mac and the Dual I/O module.

Setup: Options

The Options section is used to enable/disable various software features.

+ **Disable White Square** - Checking this box will disable the white square that appears in the Mbox window if a layer's opacity is brought up without selecting a texture (or a texture index is selected that has no associated content on the hard-drive). When disabled, nothing will show in the window - including 3D objects without textures which will be invisible. If the box is left unchecked, the square will appear when you select a texture index with any content on the hard-drive. (Refer to "Playing Back Video - Quick Start" on page 22 for an example of the white square.) To replicate the use of the white square as basic "model in white" texturing for an object, simply provide a white JPEG or PNG still image.

+ **No Diagnostics** - Checking this box will disable the Mbox automatic error-reporting images such as the icons for locked content, broken movie, etc. (Checking this box is recommended during show situations).

+ **Pixel Map Enable** - Checking this box enables the processing of pixel mapping patches and the resulting Art-Net output. (Unless you are using pixel mapping, this box should be left unchecked.)

+ **ArtPOLL Disable** - Checking this box turns off Mbox's ability to ask for and receive configuration information from Art-Net devices on the network. This information is used solely for determining the destination for particular universes of pixel mapping Art-Net data.

+ **AIFF Playback** - Checking this box enables AIFF audio playback. (The Mbox software must be restarted for this change to take effect.) This checkbox has no effect on embedded audio playback. Refer to "Embedded Audio Tracks (in QuickTime Movies)" on page 73.
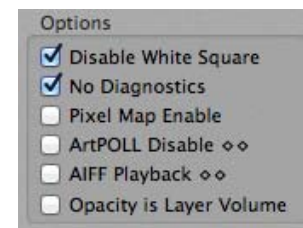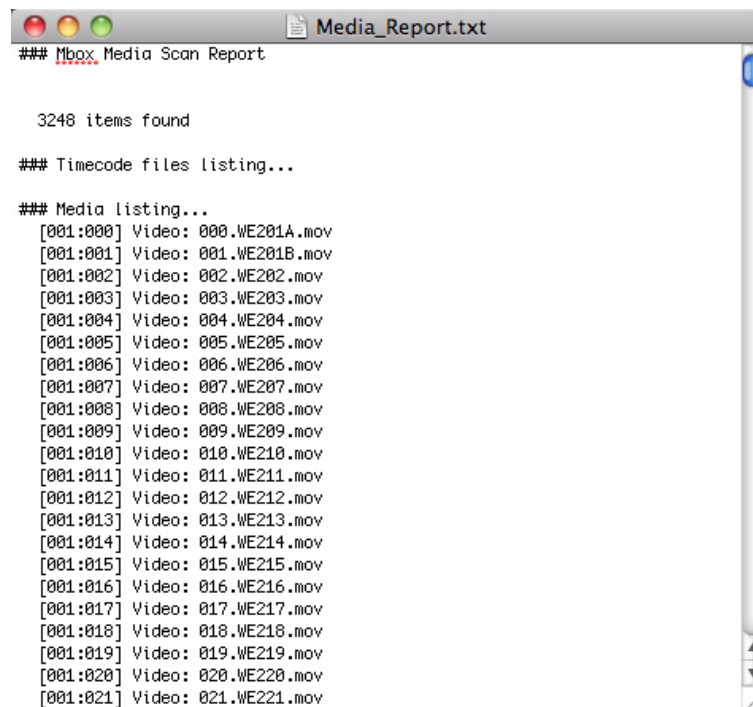
Setup: Media Log (Media Errors)

The Media Log button opens the Mbox Media Report window, which contains a list of all indexed content (textures, objects, effects, and transitions). When Mbox performs a media scan, it also looks for files with timecode links, errors, or other conflicts and reports them.
For example, two media files with the same number would be reported as an error in the log. If a file does not appear in this list, then Mbox will not attempt to play it. Therefore, this list can be used to help identify issues if media files don't play when requested.

The following is an example Media Report window:

```
### Mbox Media Scan Report


  3248 items found

### Timecode files listing...

### Media listing...
   [001:000] Video: 000.WE201A.mov
   [001:001] Video: 001.WE201B.mov
   [001:002] Video: 002.WE202.mov
   [001:003] Video: 003.WE203.mov
   [001:004] Video: 004.WE204.mov
   [001:005] Video: 005.WE205.mov
   [001:006] Video: 006.WE206.mov
   [001:007] Video: 007.WE207.mov
   [001:008] Video: 008.WE208.mov
   [001:009] Video: 009.WE209.mov
   [001:010] Video: 010.WE210.mov
   [001:011] Video: 011.WE211.mov
   [001:012] Video: 012.WE212.mov
   [001:013] Video: 013.WE213.mov
   [001:014] Video: 014.WE214.mov
   [001:015] Video: 015.WE215.mov
   [001:016] Video: 016.WE216.mov
   [001:017] Video: 017.WE217.mov
   [001:018] Video: 018.WE218.mov
   [001:019] Video: 019.WE219.mov
   [001:020] Video: 020.WE220.mov
   [001:021] Video: 021.WE221.mov
```

HEADS UP DISPLAYS (HUDS)

Overview

Mbox Mini provides Heads Up Displays (HUDs) that contain detailed information about the server. There are two methods for calling up most of these HUDs: the Master fixture control channel or the keyboard. Refer to the chart on page 135 for a listing of discrete values for the Master Control Channel. For keyboard activation of the HUDs, press one of the function keys listed below:

+ F1 - Master, Keystone, Shutter HUD
+ F2 - Texture HUD
+ F3 - Objects HUD
+ F4 - Control Universe HUD
+ F5 - Pixel Mapping Previsualization
+ F7 - Alignment Rectangles
+ F8 - Software Version HUD
+ F9 - Performance HUD
+ F10 - Timecode/Activity HUD

Example of the F4 HUD
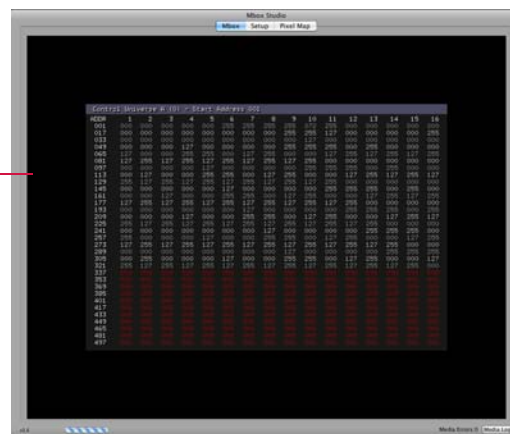
Note: On some keyboards it may be necessary to press a combination of the [fn] key and the function [F1 - F10] key since the function keys are mapped to OS-specific tasks.

Use the following guidelines when working with the HUDs:

+ Any HUD called from the keyboard will override a request for a HUD by the Master Control Channel.
+ Press [esc] to clear any keyboard HUD or press the activating key a second time.
+ HUDs activated from the keyboard can also be cleared from the control console through the use of the Master Control Channel.
+ [F5] and [F7] cycle through the modes for Pixel Mapping and Alignment Rectangles respectively.

Activity/Timecode Monitoring

Program activity and timecode input can be monitored on the display. Since the screen may be programmed to go black at certain points in time, this provides an indication that Mbox Mini is running, and also provides an on-screen indication of current timecode.

[F10] will cycle through the five display positions for the Timecode/Activity HUD: center, top-right, top-left, bottom-left, and bottom-right. The position can also be set using the Master Control Channel (page 135).

The green rectangle moves across the gray notification area from left to right, indicating program activity. When valid timecode is being received (using the Mbox Remote application), the current timecode value (HH:MM:SS:FF) will be displayed in the notification area. If timecode stops, the timecode notification will go away, but the green activity rectangle will continue to move.
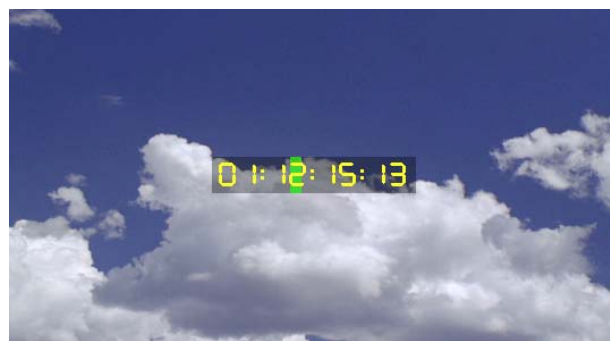
Using the HUDs

The HUDs can be very useful during programming to view the values for fixtures and parameters. When making changes to the parameters, the values will be immediately reflected in HUD screens. For example, when making changes to parameters on the Layer 1 Fixture, they can be viewed in the Layer:Texture HUD (F2).

The following screen shows Layer 1 with the default levels:

This next screen shows Layer 1 after some modifications have been made to the effects, color and position. When changes are made to a parameter, the HUD will display its value in orange. Default values are always shown in green.

All other HUDs will show the same type of changes to the parameter values. Paging through F1, F2, F3 and F4 HUDs can be helpful during programming to verify value changes, or for troubleshooting patch issues.

DISPLAY DEVICES

Connected Display Devices and EDID

Without special hardware or software, a typical computer can only output video resolutions and frequencies that are supported by the display equipment that is connected to it. Display devices communicate their required input configurations via EDID (Extended Display Identification Data), which is essentially a communication protocol used between a device and a computer.

When connecting the Mbox video outputs to a display device, you must ensure that the display device's desired resolution and frequency appear in the Mac Displays Preference window so that it will be possible to select the resolution and frequency here and in the Fullscreen Mode section of the Setup tab (refer to "Setup: Fullscreen Mode" on page 38). If the desired resolution and frequency are not available, then you must use alternate EDID data from an EDID spoofing device.

The currently selected EDID for each output is shown in the Mac Displays Preference window. Each output of the server will show a configuration similar to the example in **Figure 3-1** below. The header of each window will display the name of the EDID (e.g. PRG-MbE 60Hz as shown below).
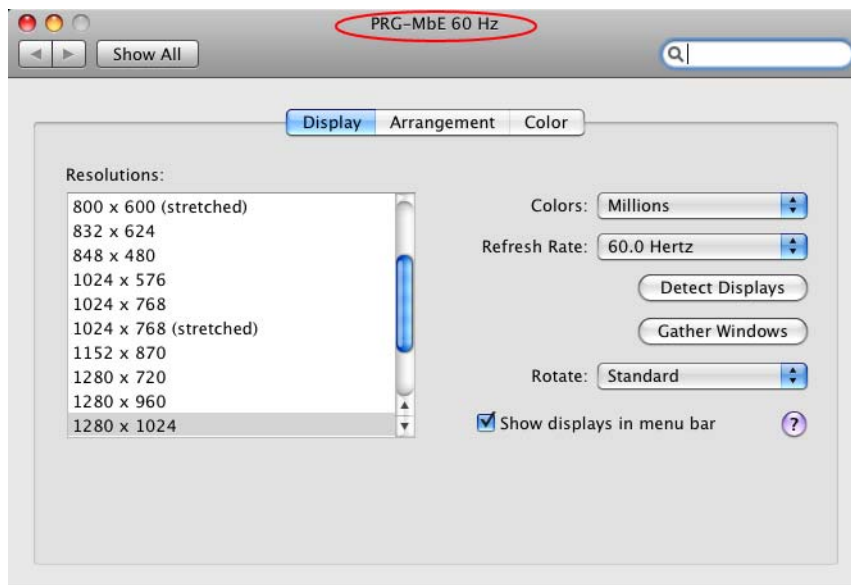
Figure 3-1: Mac Display Preferences

ENHANCING MBOX PERFORMANCE

Recommended Practices

Use the following recommended practices to enhance the performance of your Mbox system:

+ Disable any screen-savers.
+ Disable File Sharing and Remote Management (any and all sharing, in fact).
+ Turn off automatic software updates.
+ Disable Spotlight on all connected hard-drives by placing them in the Privacy table.
+ Set Energy Saver settings to disable computer and display sleep, and to stop hard-drives from being put to sleep.
+ If using a laptop with two graphics modes, disable automatic switching (use max performance).
+ Disable Time Machine.
+ Disconnect the computer from the internet.
+ Disable Expose and Spaces.
+ Turn off FileVault and Firewall.
+ Turn off Bluetooth.
+ Turn off automatic data and time setting.
+ Quit all other applications!
+ Disable unnecessary background processes.

Notes

4.

OPERATION

This chapter provides operating instructions for the Mbox Mini software.

+ PLAYMODES

+ 3D OBJECTS AND DIGITAL GOBOS

+ ANIMATED TEXTURES

+ DRAWMODES

+ LAYER BLENDING MODES

+ LAYER COPY

+ TEXT FILES

+ SCRIPTS

+ AUDIO

+ SYNCHRONIZATION

+ ALIGNMENT RECTANGLES

+ EFFECTS AND TRANSITIONS

+ PROJECTION MAPPING

+ PIXEL MAPPING

PLAYMODES

Basic Modes

Loop (Forward & Reverse) - Looping playmodes will play the movie from In-Frame to Out-Frame (Forward) or Out-Frame to In-Frame (Reverse) in a looping fashion. When the layer's opacity is reduced to 0, the playhead continues to mark time, meaning that when opacity is restored, the playhead will be at a later point in time.

Once (Forward & Reverse) - Once playmodes will play the movie from In-Frame to Out-Frame (Forward) or Out-Frame to In-Frame (Reverse) but will pause on the final frame rather than looping. When the layer's opacity is reduced to 0, the playhead continues to mark time, meaning that when opacity is restored, the playhead will be at a later point in time.

Modifiers:

+ **Pause when Opacity = 0** - When the layer's opacity is reduced to a level of 0, the playhead stops. When opacity is restored, playback resumes with the playhead at the same point it was when opacity was set to 0.

+ **Pause and Reset when Opacity = 0** - When the layer's opacity is reduced to a level of 0, the playhead stops. When opacity is restored, playback resumes at the In-Frame/Out-Frame - depending on the direction of playback.

Bounce/Random

Bounce (Forward & Reverse) - Bounce playmodes will play the movie in the selected direction initially, but playback will switch to the the opposite direction when the last frame is reached. (This is also know as "ping-pong.")

Random - A pseudo-random frame of the movie from between the In and Out-Frames will be displayed. This mode is pseudo-random because using the same movie and playback settings on more than one layer/server will cause the same frame to be played at the same time on both layers/servers. This is done so as to allow for playback sync between multiple servers in a blended arrangement.

Timecode Sync

Strict Lock - Used with a timecode (LTC or MTC) input. Movie playback waits on In-Frame for matching timecode value, at which point playback is locked to timecode. If timecode stops, the playback stops. If timecode jumps (forward or backwards) playback will jump accordingly.

Sync then Freewheel - Used with a timecode (LTC or MTC) input. Movie playback waits on In-Frame for matching timecode value, at which point playback is started but ignores timecode (freewheels) until the movie or playmode is changed on the layer.

Jam Sync - Used with a timecode (LTC or MTC) input. Movie playback waits on In-Frame for matching timecode value. If timecode continues playback is locked to timecode. If timecode goes away, playback continues (freewheels). If timecode returns, the playhead will jump to the point of the movie matching that timecode.

Layer Sync

Slave - Used with one of the Layer Sync Master modes on another layer/server. A Slave layer pauses on its In-Frame and listens for sync signals on its selected Sync Stream. If sync is received, the layer plays the same relative frame as the Master, and at the same speed.

Master Forward Loop - Used with the Layer Sync Slave mode on another layer/server. A Master Forward Loop layer plays between In and Out-Frames, loops at the Out-Frame, and transmits a sync signal on the selected Sync Stream.

Master Forward Once - Used with the Layer Sync Slave mode on another layer/server. A Master Forward Loop layer plays between In and Out-Frames, pauses at the Out-Frame, and transmits a sync signal on the selected Sync Stream.

Kiosk

Non-Looping, Pause on Last Frame of Last File - Playback between In and Out-Frame begins with the initially-selected file in a folder, automatic sequential transitions (using layer's transition type and time) happen until the Out-Frame of the last movie in the folder is reached. At which point the movie will pause on the Out-Frame until the movie or playmode is changed.

Non-Looping, Loop Last File - Playback between In and Out-Frame begins with the initially-selected file in a folder, automatic sequential transitions (using layer's transition type and time) happen until the Out-Frame of the last movie in the folder is reached. At which point the movie will loop back to its In-Frame continuously until the movie or playmode is changed.

Non-Looping, Fade Out Last File - Playback between In and Out-Frame begins with the initially-selected file in a folder, automatic sequential transitions (using layer's transition type and time) happen until the Out-Frame of the last movie in the folder is reached. At which point the movie will fade to transparent on the Out-Frame, remaining invisible until the movie or playmode is changed.

Looping - Playback between In and Out-Frame begins with the initially-selected file in a folder, automatic sequential transitions (using layer's transition type and time) happen until the Out-Frame of the last movie in the folder is reached. At which point the movie will loop to the In-Frame of the first movie in the selected folder.

SMPTE - A valid movie folder/file must be selected to initiate playback. This movie will stay paused on its In-Frame until timecode reaches the start of any movie in the folder. At which point the incoming timecode value will trigger the playback of the movie with the matching time.

Guidelines:

+ The two non-SMPTE Kiosk modes use the layer's current crossfade type and time when switching from one piece of content to the next. These two modes require the first desired piece of content in the folder to be selected.

+ The SMPTE Kiosk mode requires any valid SMPTE-linked piece of content in the folder to be selected; this should typically be the first piece of content desired. Folders for use with SMPTE Kiosk mode should <u>not</u> include any non-SMPTE linked content. It is also a good idea to make the first and last frame of any SMPTE-linked content black to avoid an undesirable paused frame as the layer waits for the appropriate SMPTE to initiate playback.

+ The SMPTE Kiosk mode requires the use of the MediaTimeCodes.plist for assignment of start times for the individual files (refer to "Linking Files to Timecode (Method 1)" on page 74. It is not compatible with the TC DMX Controls.

Restart

Restart Movie from In-Frame - This playmode always returns the playhead to the current In-Frame point. The Playmode value should be returned to its prior value after a restart is triggered.

📄 For a complete channel map, refer to "Play Mode" on page 137.

3D OBJECTS AND DIGITAL GOBOS

About 3D Models and Digital Gobos (Objects)

3D Model and Digital Gobo files are arranged into individual folders within the /Mbox/Models folder.
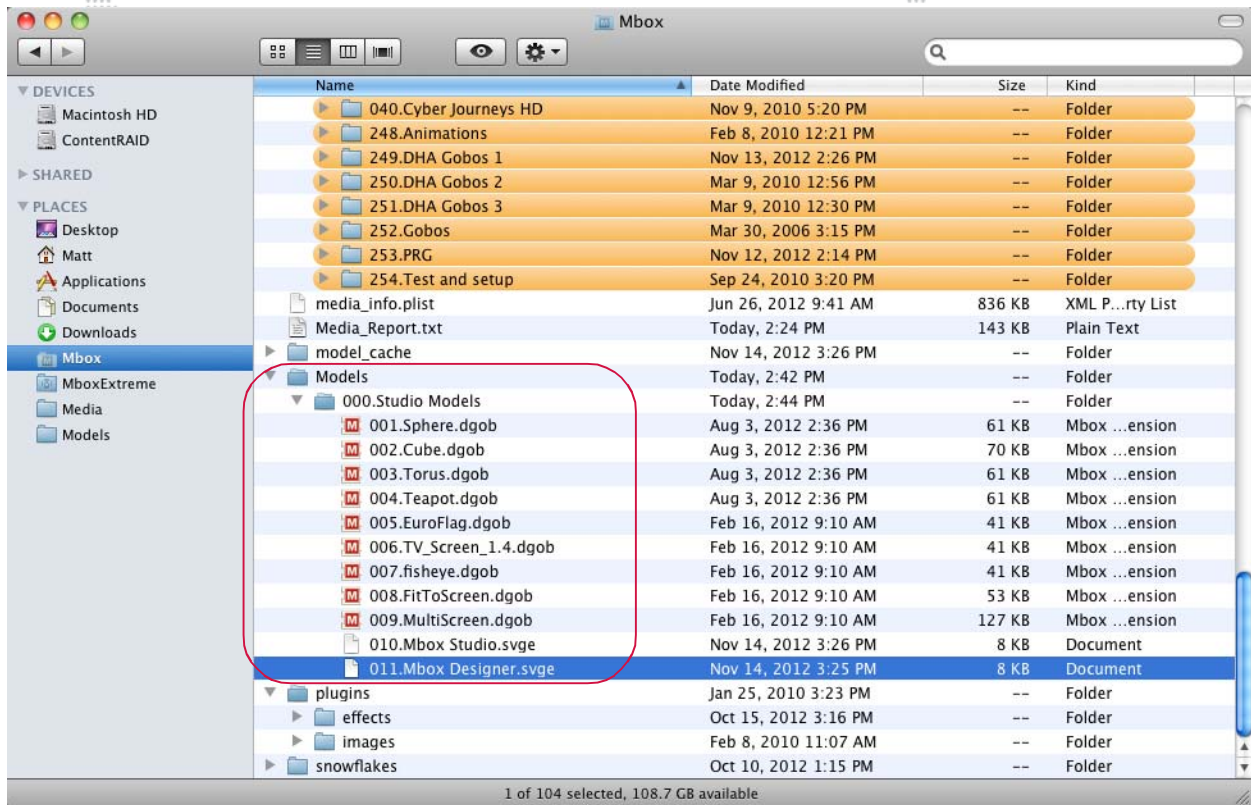
Lighting

All 3D models and digital gobos require some amount of lighting to be properly visible. Mbox Mini uses default lighting settings for 3D objects, which is not adjustable. You may turn on full ambient lighting for any layer by using Drawmode 1 (refer to "Using Drawmodes" on page 63).

Vertex Array Files (Compiling)

When displaying a 3D model or digital gobo file on the screen, that file is read from the hard-drive just like the 2D content. But loading model and digital gobo files in their original format (.obj, .svg) and converting them to be used within the Mbox environment can be a slow process. Instead, Mbox Mini generates a new version of these files in the Mbox/Models folder using a proprietary format. The resulting vertex array files have been optimized to allow for shorter load times when the files are requested for display. The process of creating the vertex array files is called "compiling."

All appropriate files in the *Models* folder will be compiled automatically the first time Mbox Mini is started (and successfully locates them). Thereafter, files in the *Models* folder will only be compiled if they are new or if they have been modified since the last time Mbox Mini was used. Compiling all of the stock files in the *Models* folder takes some time, but this task occurs in the background, allowing all other functions to be available during this process. If models or digital gobos that are still waiting to be compiled are requested, Mbox will display a substitute image until the requested file has been compiled. If a file that hasn't been compiled yet is requested, Mbox will display the "Waiting for Gobot" gobo (shown at right).
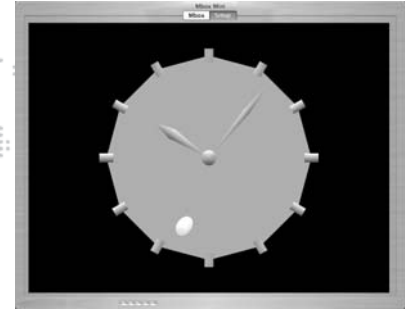
Waiting For Gobot

Note: If you place a large number of new files in the *Models* folder, don't expect them to be available as soon as the software launches, but once compiled, the files will be available immediately.

Compiled object files are automatically placed in the *Mbox/model_cache* folder. These files are given a numerical prefix and the .vtxa (vertex array) suffix. These vertex array files cannot edited or renumbered. Any additions, deletions, or renumbering of model or digital gobo content should be done in the *Models* folder rather than in the *model_cache* folder.

Normalization

By default, both model and digital gobo files are "normalized" when they are compiled. The Mbox vertex array compiler performs this task to ensure that the files appear at a usable scale rather than being too large or too small. Given the wide variety of applications that can be used to create models or digital gobos, an object that appears to be just right when created might appear to be gigantic or tiny when displayed by Mbox. Different applications may use different units of scale – pixels, inches, meters – but each of these units of measure will be saved as a generic unit in an .obj or .svg file.

Therefore, when compiling, Mbox's default action is to normalize all models and digital gobos. Normalization is accomplished by scaling objects to fit into a bounding cube that is 480px on each side. Objects are scaled up or down so that their largest dimension is 480px and their other dimensions are then scaled proportionally.

To stop the compiler from normalizing an object, the suffix ".noscale" can be added to its name before the file is compiled and the file will not be normalized when it is compiled. If this option is used, one generic unit will equal 240 pixels in the Mbox 3D environment. Example:

a file named - **001.MyObject.obj** should be renamed as - **001.MyObject.noscale.obj**

To create an object with a scale factor (relative to other files), use the ".scale" switch. In the following example, the second file would appear at half the size of the original:

a file named - **023.DigiGobo.svg** would be renamed as - **023.DigiGobo.scale_0.50.svg**

If you create an object that is scaled to match a particular piece of 2D content (which is measured in pixels), and you want Mbox to treat generic units as pixels, add the suffix ".pixels" to the file's name before it is compiled. Example:

a file named - **001.NewObject.obj** should be renamed as - **001.NewObject.pixels.obj**

In this case, an object that is 200 generic units tall will appear as 200 pixels tall within Mbox. And a piece of 2D content that is 200px high would fit exactly when applied to that object.

Creating Custom Digital Gobos

Overview

Custom digital gobos can be created in two formats: SVG and AI (Adobe Illustrator®). Adobe Illustrator is the best application for creating custom digital gobos because it provides the best toolset for features that Mbox supports. SVG files are the most compatible and they provide better functionality than AI files.

The Mbox vertex array compiler pays attention to "object fill" information in these file types. Path and Stroke information is ignored, unless special steps are taken with the file. (Refer to the Adobe Illustrator documentation for more information on path, fill, and stroke.)

Scale-To-Fit vs. Arbitrary Fit

Before beginning, determine the intent of the digital gobo. If you create an SVG gobo file, you have two options for how a texture is mapped to the digital gobo:

+ **Scale To Fit** - This will have a texture stretched (or squashed) to fit onto the digital gobo, provided you have a shape that you want to apply the texture to.

+ **Arbitrary Fit** - This will have a texture applied to a digital gobo with a specific fill factor given to the texture - if you are using a digital gobo as a texturable masking aid or if you don't want the texture scaled. If you create an AI gobo file, you are limited to the Scale To Fit option, and the procedure is slightly different.

Creating a Digital Gobo with Adobe Illustrator CS5

The following steps provide a general guideline for creating custom digital gobos using Adobe Illustrator CS5. Note that older versions of Illustrator may require slightly different steps to achieve the same results. For more information regarding the use of Illustrator, consult the Illustrator Help documentation included with the application.

Step   1.   Start a new document and name it appropriately.   The size of the artboard and orientation of the document are not critical, nor is the color space (RGB or CMYK). Basically, just make sure the document is large enough to hold all of the objects.
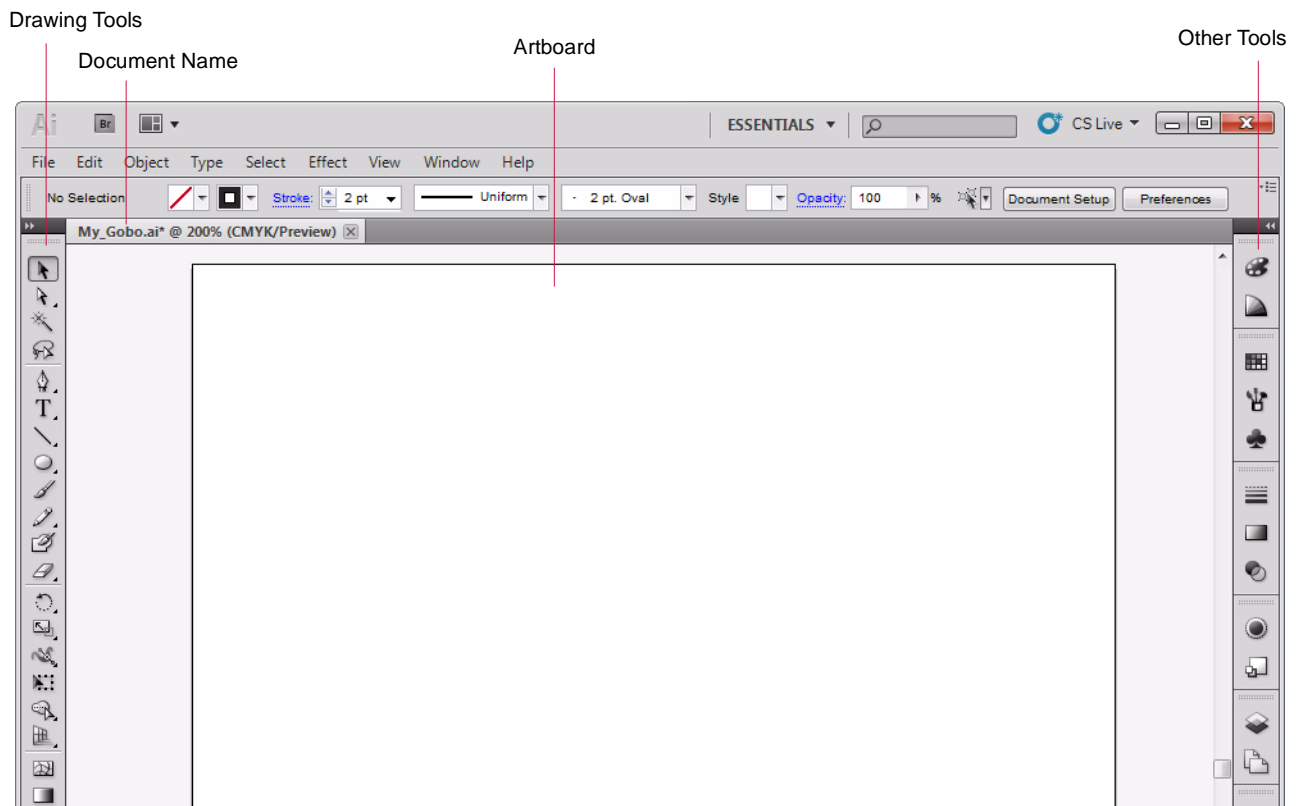
Step 2. Use the various drawing tools to compose the gobo. The most useful tools are the objects (ellipse, rectangle, polygon, or star), the paintbrush, the linear tools (line, arc, spiral), text tool, and the pencil. All of these tools, except the line tool, will create a filled object. Filled objects will be compiled as solid areas in Mbox and will be the areas that accept a texture, if one is applied.

Text
Lines
Objects
Paint Brush
Pencil

Rectangle Tool   (M)
Rounded Rectangle Tool
Ellipse Tool   (L)
Polygon Tool
Star Tool
Flare Tool

Tool Options
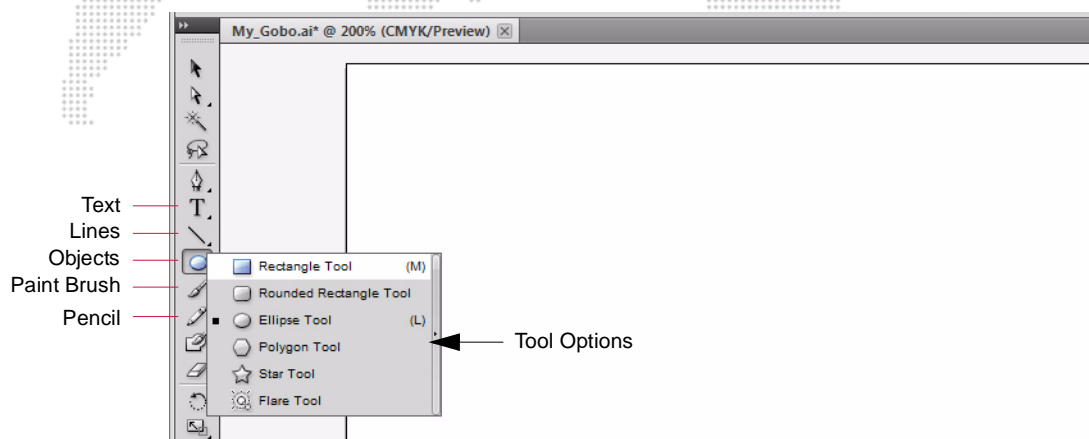
Step 3. To make an outline of an object, turn off the object's fill and then expand the object. To turn off the fill, select the object in question and then make sure the Fill/Stroke portion of the toolbar looks like the example below.

Filled Object   Outlined (Stroked) Object   Fill/Stroke Color Options

Fill/Stroke Setting
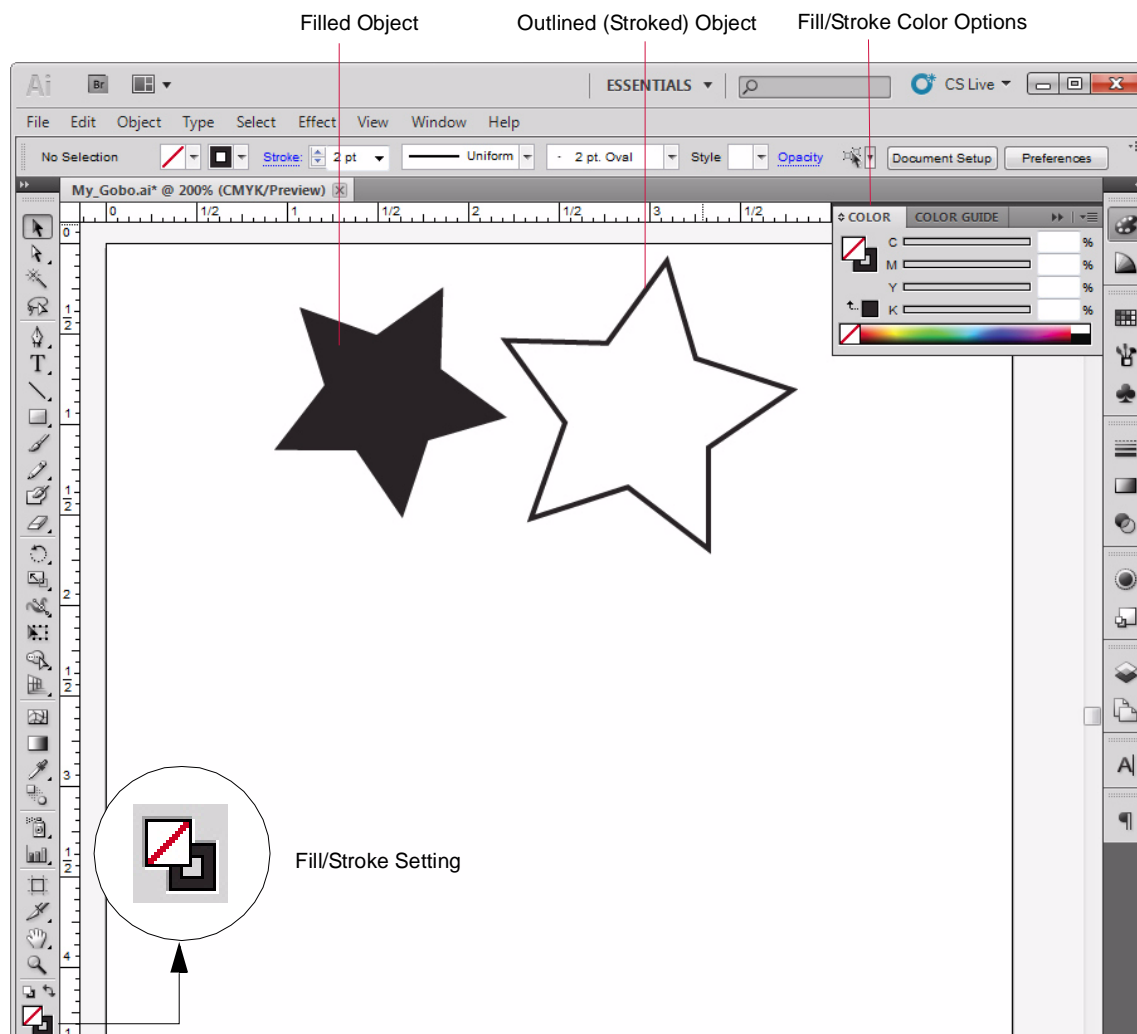
Step 4. To expand an object, first select the object, then choose **Expand** from the **Object** menu. This will open a pop-up window. You don't need to change any of the options in the window, just press **OK**. Add more objects as desired. (You can use multiple layers, if required.)
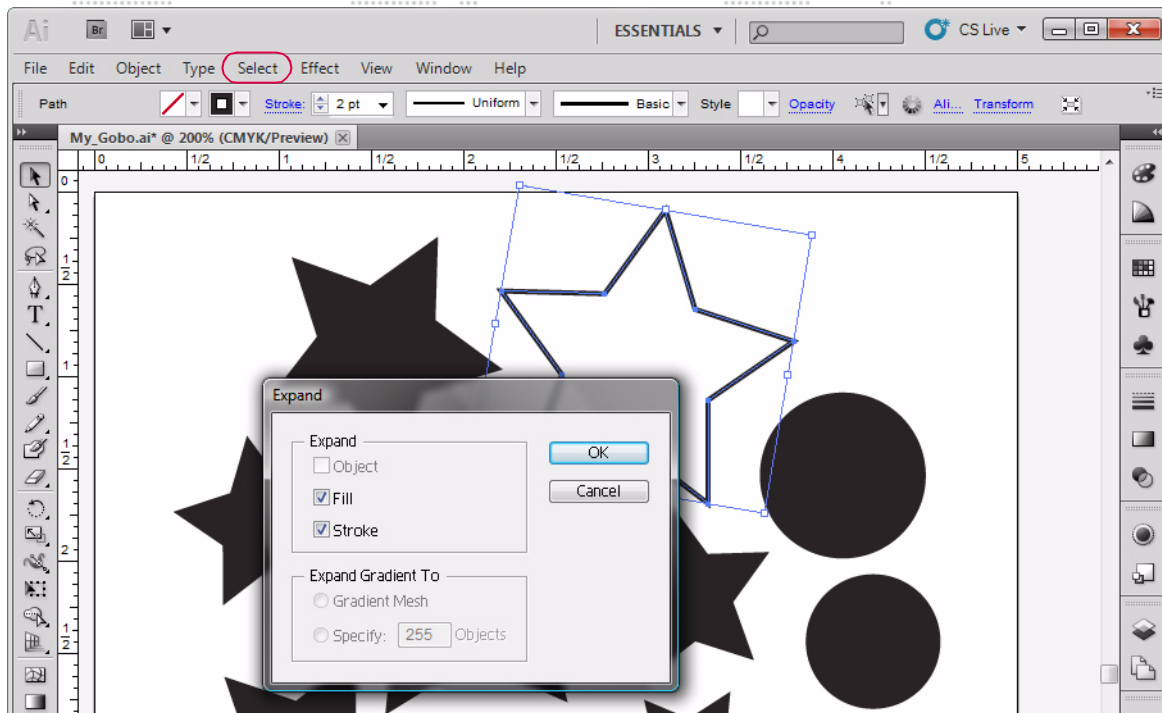
Step 5. Use the text tool to create text as desired. Use the Character options to set font and size.

Character Options

Step   6. Text is a special case, and must be converted out "outlines." Switch to the Selection Tool and select the text. Select "Type" in the menu bar, and then select "Create Outlines." You cannot make outlined text (stroke only), it has to be filled.
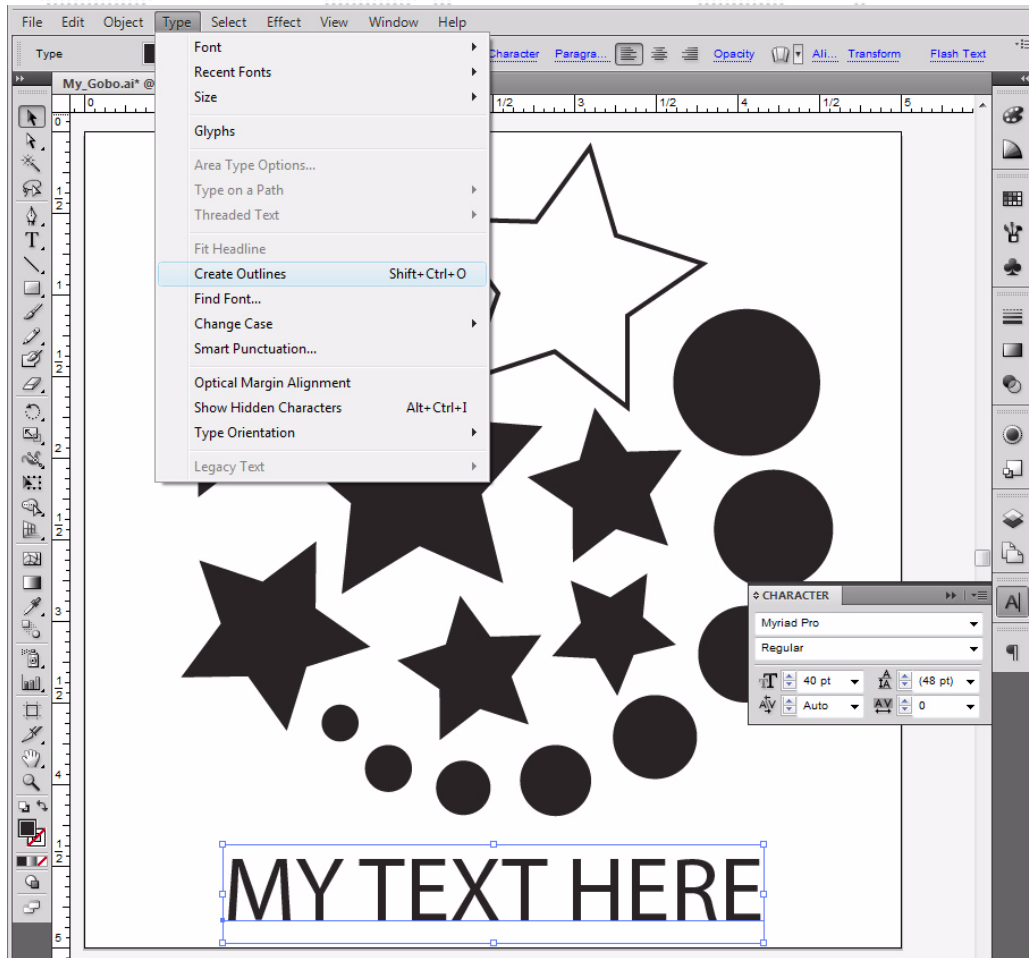
Step   7. If you are creating an SVG gobo, you need to apply a crop area to the document. The size and location of this crop area will determine whether your digital gobo will fall under Option 1 or Option 2. The texture applied to the gobo will fill the crop area when the gobo is used in Mbox. Note: It is not recommended that the crop area be sized or positioned so that any portion of an object fall outside the crop area.

   a. Option 1: If you just want to apply a texture to your gobo so that the texture completely fills the gobo, then create the crop area so its borders are just outside the edges of all of the objects in your composition.

   b. Option 2: If you have a texture that you do not want scaled (or want scaled to a specific size) and only want to show that texture in the positive areas of the gobo without any further scaling, then make the crop area the same as your content and arrange the objects in the file inside that crop area. In this case the gobo acts as an inverted mask since the result ends up displaying the unscaled texture on the the positive areas of the SVG file.

   If you are creating an AI gobo, you can't use crop areas, and the applied texture will be stretched or squashed automatically to cover all of the objects in the file. You can use rectangles with no dimensions (0px x 0px) to affect how the texture is applied by using them to define the corners of an imaginary crop area. To add a rectangle with no dimensions, select the rectangle tool and click and release on the drawing area. In the dialog box that opens, enter 0 for the horizontal and vertical size of the rectangle. You won't be able to see these objects in Mbox, but they will affect how the texture is applied.

Step 8. Before you saving the file, consider where the center of the file should be. By default, the center of a digital gobo will appear in the center of the Mbox window and the gobo will rotate around that center point. If you save the file as an AI file, you can set the center by modifying the origin of the file in Illustrator. To modify the origin of a file, click and hold on the small, dotted set of crosshairs at the top-left of the document window then drag the crosshairs to where you want the origin to be. For optimal results, the new origin must be within the boundaries of one of the objects in the file. Note: You must have rulers turned on to see the origin and crosshairs.
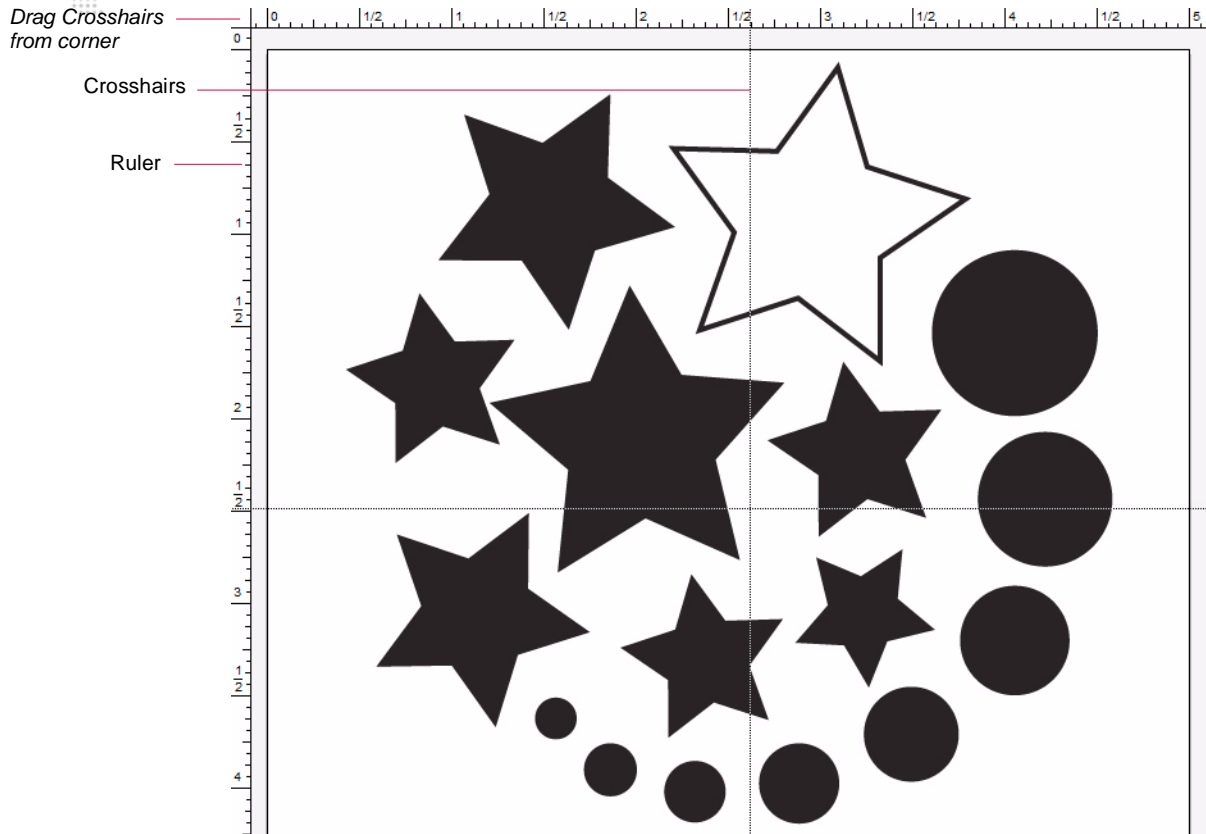
If you save your file as an SVG file, then the rotational center of the file can either be the center of the crop area, or you can add a rectangle with no dimensions which then defines an alternate center point. This 0_{px} x 0_{px} rectangle can be either inside or outside of other objects in the file and inside or outside the crop area.

Step 9. Save file, choosing either the SVG or AI format. Both file types will prompt a secondary pop-up window to set options when saving the file. When saving as an SVG file, you do not need to modify any of the default values. When saving as an AI file, you need to make sure that "Use Compression" is not checked.
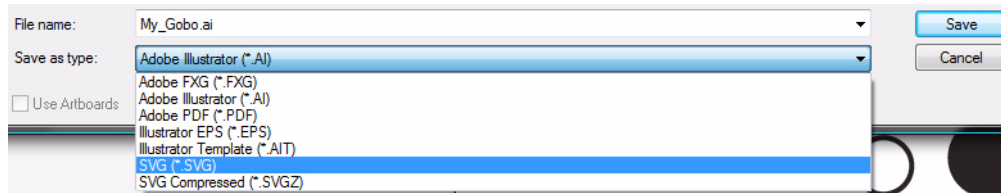
Step 10. When creating an SVG file, change the file extension from .svg to .svge to allow Mbox Mini to generate thumbnails of the SVG file.

Animated Digital Gobos

Mbox animated gobos are similar to SVG gobos, but they have special functions. Animated gobos, saved as .dgob files, are located in the */Mbox/Models* folder. Unlike normal digital gobos, these files do not need to be compiled so there will be no corresponding .vtxa files. Like other digital gobos, the animated gobos are treated as objects and some of them require lighting.
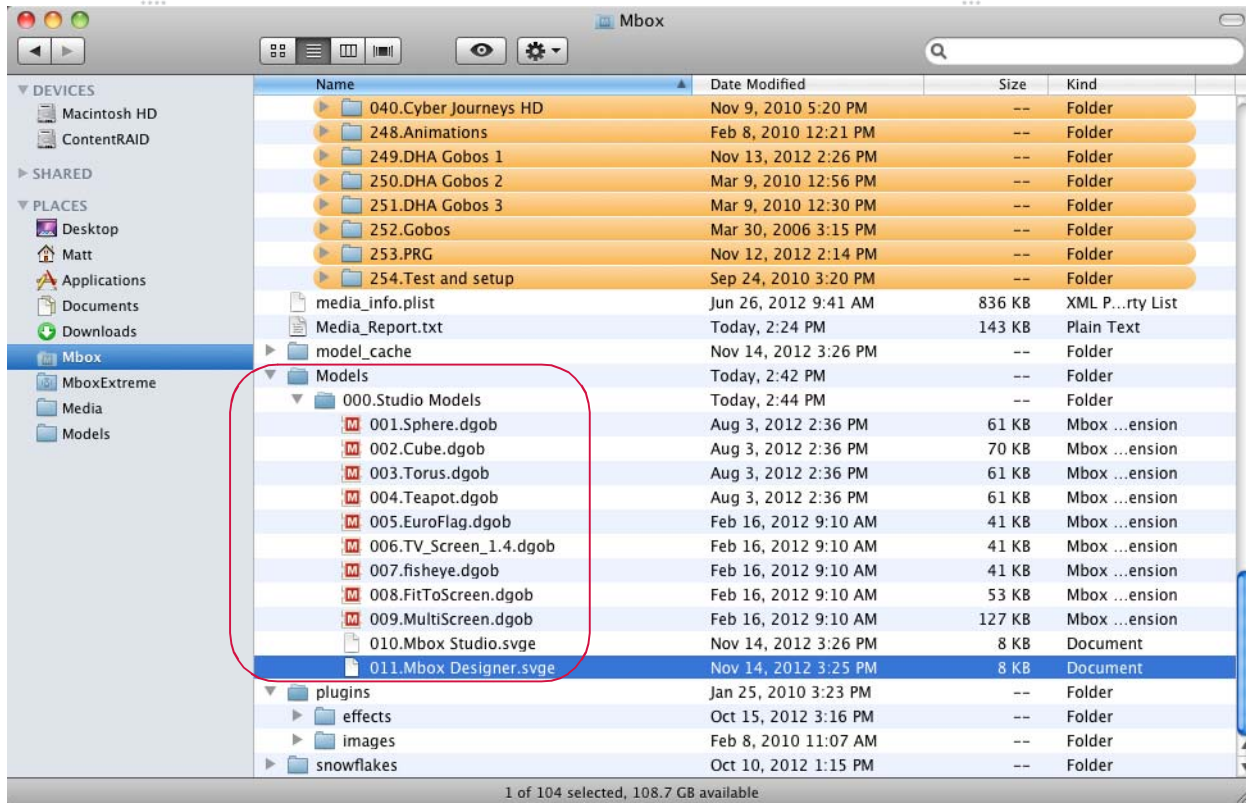
EuroFlag

The EuroFlag animated gobo is very basic. The layer's texture is applied to each star.

TV Screen v1.4

The TV Screen animated gobo allows a texture to be placed onto a plane in 3D space and manipulated it as a 3D object. The texture can rotate and spin on all three X, Y, and Z axes. (A texture on its own will only rotate on the Z axis.) The TV Screen will automatically light itself and Drawmode 1 does not have to be set for it to be visible.
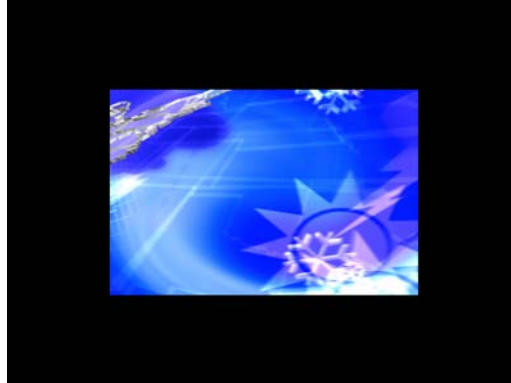
Fisheye

The Fisheye gobo is used for the conical geometric correction necessary when using a very wide angle lens and projecting into the inside of a translucent sphere.
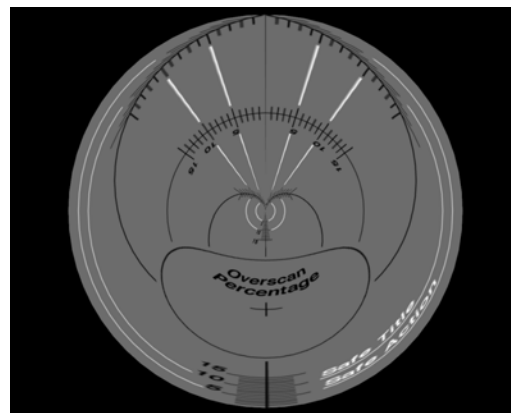
Fit To Screen

The Fit To Screen animated gobo can be used to force any content to fill the screen. The image is scaled so both width and height are >= screen size (i.e. the smaller dimension is scaled to fill, so the larger dimension overfills the screen and is cropped). The image aspect ratio is not affected.

MultiScreen Gobo

The MultiScreen animated gobo is used with the Mbox Pixel Remapping features to allow placement of content onto a predefined structure that can receive external data (Art-Net) to control both the position and arrangement of the content on the structure, and also the structure's position in 2D and 3D space. The MultiScreen gobo is intended for use with complicated or creative video arrangements that require pixel remapping, or that involve motion (scenic tracking) during playback. A MultiScreen Editor provides the toolset for creating and editing the MultiScreen.plist file, which the MultiScreen animated gobo uses as its instruction set. For complete instructions on using the MultiScreen Gobo, refer to the

ANIMATED TEXTURES

Using Animated Textures (Animations)

Animated Textures are found in folder 248 of the *Media* folder. They are similar to the Animated Gobos in the *Models* folder, but are controlled differently and cannot have a texture (movie, still image, etc.) applied to them. Each animated texture has white areas and transparent areas. The white areas can be colored using the RGB controls on the layer. The layer's movie playback controls are retasked to set the parameters for the animation – speed, direction, etc.
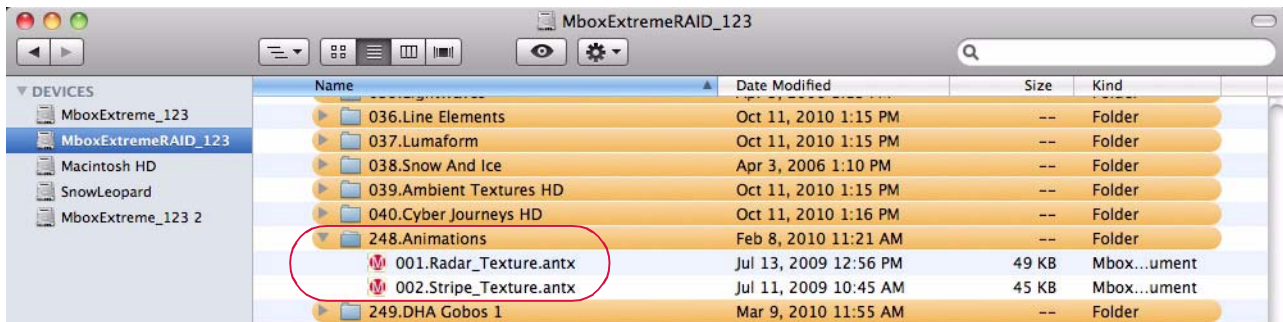
Playback Controls:

+ **In Point** – thickness of objects (lines, rays, etc.). A value of 0 gives a default thickness.
+ **Out Point** - number of objects. A value of 65535 gives a default number of objects.
+ **PlaySpeed** - speed and direction control. A value of 127 equals no movement.
+ **Frame Blending** - decay. A value of 0 equals no decay, 255 is full decay.
+ **PlayMode** – see chart below.

PlayMode	Radar_Texture (248.001)	Stripe_Texture (248.002)
FWD Loop	Sweep	Left-Right scroll
REV Loop	Bounce	Bounce
FWD Once	Once	Once
REV Once	Scrub	Scrub
FWD Bounce	Mirrored sweep	Top-Bottom scroll
REV Bounce	Bounce	Bounce
Random	Once	Once
FWD IPR	Scrub	Scrub
Scrub In	Mirrored fan	Mirrored L-R
Scrub Out	Bounce	Bounce
FWD Loop XF	Once	Once
Rev Loop XF	Scrub	Scrub
FWD Loop IPR	n/a	Mirrored T-B
Timecode sync	n/a	Bounce
Layer Slave	n/a	Once
Layer Master	n/a	Scrub

📄 For a complete channel map, refer to "Play Mode" on page 137.

DRAWMODES

Using Drawmodes

The Drawmodes feature provides control over how a 3D object is lit and how it interacts with other layers. Drawmode has its own control channel with several modes:

- + 0 = Idle
- + 1 = Light
- + 2 = Cut Stencil
- + 3 = Cut plus Light

- + 4 = Cut and Draw
- + 5 = Cut, Draw plus Light
- + 6 = Draw Thru Stencil
- + 7 = Draw Thru plus Light

- + 8 = Draw Onto Stencil
- + 9 = Draw Onto plus Light

📄 For a complete channel map, refer to

Example:

Mode 1, Light, automatically raises the ambient lighting level to 100% for the selected layer only. This control works with both gobos and 3D objects and enables objects (and their applied textures) to be seen without affecting the lighting on other layers.

To achieve the results shown in the images to the right, do the following:

1 Set the Master intensity to 100% and then, on Layer 1 only, with all other attributes set to default values, set the opacity to 100%. (You should see a white square.)

2 With Layer 1 still selected, choose Object 10 from the Objects folder. (The screen should be black - you cannot see anything.)

3 At Drawmode control, change from 0 to 1. The gobo should now be visible. Feel free to experiment with the gobo at this point. Because it is essentially a 3D object (but with no thickness on the Z axis), you can do pretty much everything to it that you can do to a 3D object: scale, rotation, spin, and X/Y/Z position. Most importantly, you can apply a texture.

4 Image 4 is the same gobo with a texture applied. In this case, it is acting like a mask; the texture is applied onto the gobo, but it appears you're looking through the gobo to a layer behind it. However, you're only using one layer not two. You can set the Drawmode back to 0 and try out the other lighting fixtures to see how they affect the gobo.
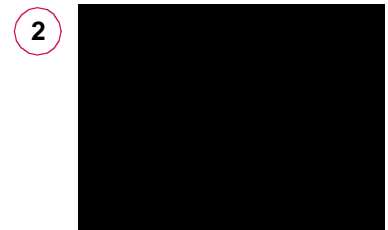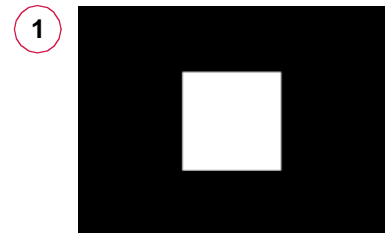
The Stencil

To explain the Drawmodes other than "Light" requires an explanation of the "Stencil" and what it means to cut it, draw onto it and draw through it. The Stencil is an imaginary layer that can be cut, as if cutting holes in a piece of paper. The Stencil can then be used as a tool to tell other layers how their textures are applied: either onto the areas where the stencil is whole or the areas where the Stencil has holes in it.

There are two important things to note:

1) When using a layer to cut the Stencil, the holes that it makes in the Stencil will only affect higher numbered layers. For example, if Layer 3 is used to cut the stencil, that cut will not affect Layer 1 or 2 no matter what Drawmode they are set to.

2) For useful results, the layer used to cut the Stencil must have some transparency. In this case, use a 3D object or gobo or apply a texture effect to the texture on that layer to give it transparency in some areas. If using an unaltered rectangular image or movie clip to cut the Stencil, it will only create a rectangular hole in the Stencil.

The following example shows a blue rectangle that represents the Stencil and a red gobo that will cut the Stencil:
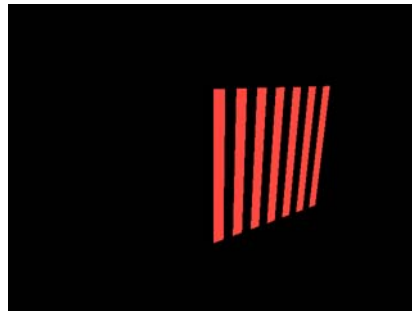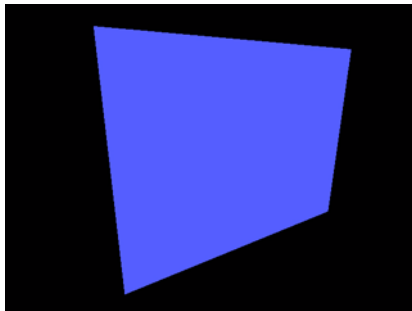
The following example first shows the two together and then shows the holes being cut into the Stencil by the gobo:
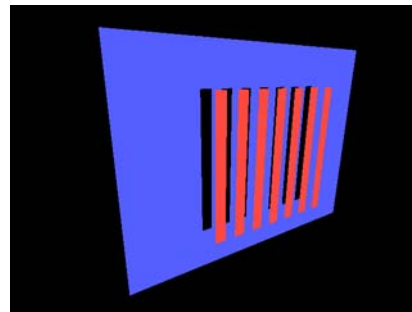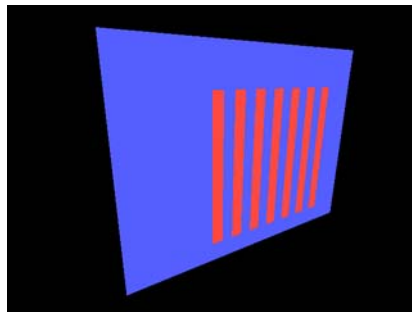
Finally, the following example shows the Stencil on its own showing the holes cut into it:
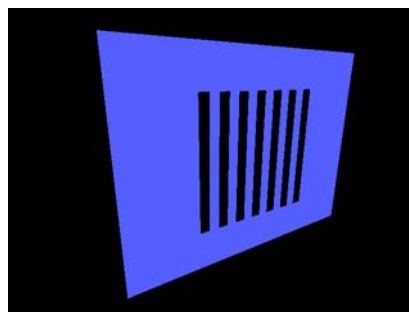
The following shows an example of using the Stencil to affect a higher numbered layer. The left image shows a digital gobo on Layer 1 in Drawmode 1 (Light) so that it can be seen. The right image shows a movie clip on Layer 2 in Drawmode 0 (idle). The movie on Layer 2 obscures the gobo on Layer 1.

In the example below, on the left, Layer 2 has been set to Drawmode 6 (Draw thru Stencil) and on the right it has been set to Drawmode 8 (Draw onto Stencil). In both cases, the gobo on Layer 1 has been set to Drawmode 2 (Cut) otherwise it would not have any effect upon the Stencil. Note that the lighting on the gobo is off, since the Drawmode has been changed.
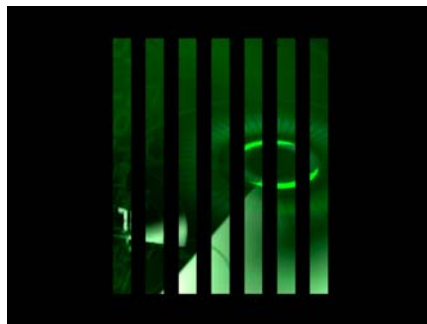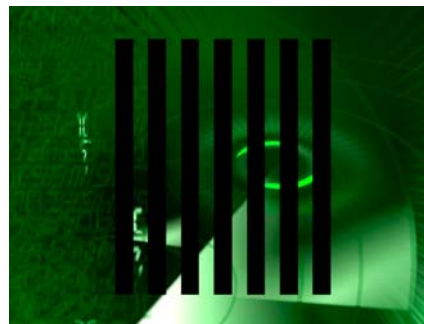
draw thru stencil   draw onto stencil

It is important to realize that when a layer's Drawmode is set to 6, the term "Draw thru Stencil" really means: draw this layer's texture only where the holes are in the Stencil. In the opposite case, if a layer's Drawmode is set to 8, the term "Draw onto Stencil" really means: draw this layer's texture only where the Stencil does not have holes.

By setting Layer 1 to Drawmode 5 (Cut and Draw plus Light) and applying a texture to Layer 1, Layer 1's texture can be shown instead of the black areas seen in the right-hand image above.

In the example below, Drawmode 5 causes the shape of Layer 1's gobo to be cut into the Stencil. Also, Layer 1's texture is applied to the gobo and, therefore, that texture is shown in the holes in the Stencil. In this particular example, the same result could be achieved without using Drawmodes by placing the green clip on Layer 1 and placing the textured gobo on Layer 2 with Layer 1's Drawmode set to 1.
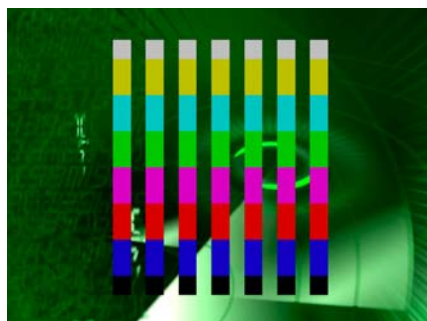
The benefits of the various Drawmodes become more apparent when the holes cut into the Stencil are used to affect more than one layer. Interesting results can be achieved by using a movie with a transparency effect to cut the Stencil. In the example below, the left-hand image shows a movie playing on Layer 1, and on the right, that same movie with the White Alpha Effect (056) turned on.
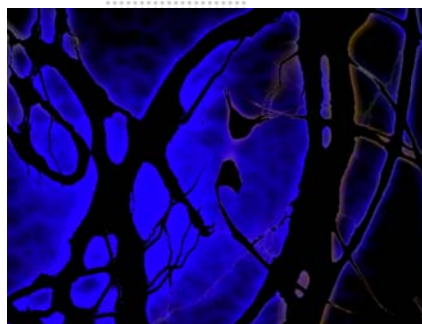
In the following example, Layer 1 is using Drawmode 5 and a movie is placed on Layer 2, shown with Drawmode 6 on the left and then with Drawmode 8 on the right.

draw thru stencil   draw onto stencil
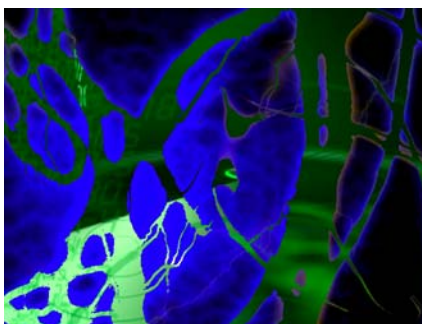
![video icon] Training Video: Mbox Media Server Training Video Chapter 10: Media Server Concepts

LAYER BLENDING MODES

Using Layer Blending

There are seven layer blending modes. Many of these layer blending modes have been chosen to best replicate common modes found in commercial image editing software (e.g. PhotoShop®, After Effects®). The modes and their values are:

+ **0 - Default Overlay**: no layer blending
+ **1 - Additive**: layer's colors are added to underlying colors (blacks appear transparent).
+ **2 - Screen**: similar to above, with less of underlying color (blacks appear transparent).
+ **3 - Multiply**: multiplies a layer's color with underlying color (blacks appear opaque).
+ **4 - Subtractive**: layer's colors are subtracted from underlying colors (blacks appear transparent).
+ **5 - Exclusion**: underlying colors are inverted where layer color is lighter; layer's colors are then added to underlying colors (blacks appear transparent).
+ **6 - Invert Subtractive**: underlying colors are subtracted from layer's colors (blacks appear opaque).
+ **7 - Invert Additive**: layer's colors are inverted and are added to inverse of underlying colors (blacks appear transparent).

The blending modes are accessed using the Drawmode control on each layer. Their discrete values have been structured so that the pre-existing drawmode controls can still be used in combination with the new blend modes. As noted in "Using Drawmodes" on page 63, the Drawmode control uses values between 1 and 31 to select combinations of functionality. These original values can be added to the blend mode values noted above to in order to combine the functions. For example, Drawmode "Draw thru stencil" has a value of 6. Blend Mode Exclusion has a value of 128. To use these two items together, set the drawmode channel to a value of 6 + 128 = 134.
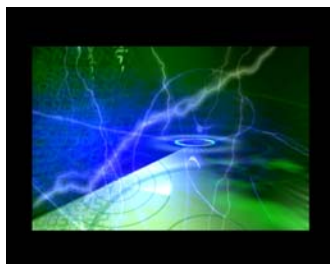
| Overlay (default) | Additive | Screen | Multiply |

| Subtractive | Exclusion | Invert Subtractive | Invert Additive |
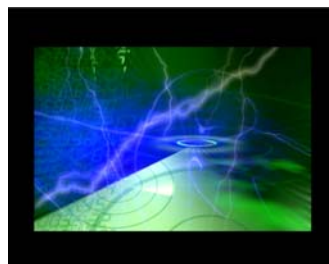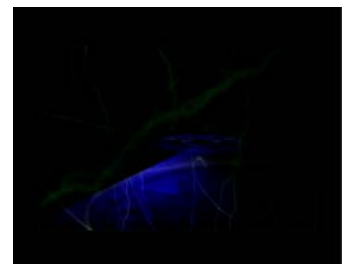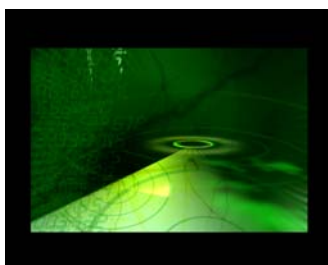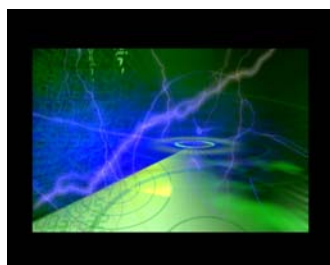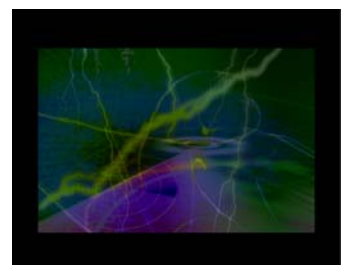
LAYER COPY

Using Layer Copy

At times, there may be a need to play the same movie content on more than one layer at the same time. The same content can be requested on multiple layers, but doing so causes the content to be read from the hard-drive more than once, as well as being decompressed more than once. This is a waste of resources if you intend to display the same frame of the same movie on multiple layers at the same time. The Layer Copy function provides a way to copy the content from one layer to another and to receive the source layer's effects and transitions.

There is one mode of Layer Copy: FX+. In this mode, the source layer sends its effects and transitions to the destination layer. The destination layer can add more effects via its own controls (allowing up to four effects to be visible on a destination layer).

Layer Copy functionality is selected as if it were a piece of content in the media folder. Folder 255 is selected, and the file number attribute controls which layer is being copied and which mode of copying takes place.

Folder 255 / File #	Mode	Layers	Note
121 - 124	FX+	1 - 4	File numbers 121-124 correspond to the FX+ mode for Layers 1-4. i.e. 121 = Layer 1 FX+ copy mode, 122 = Layer 2 FX+ copy mode.

Guidelines:

+ Layer Copy does not work with transition 255 (Object Dissolve).

+ A layer cannot copy another layer that is already copying a third layer.

Syphon

Mbox Mini includes the Syphon framework, which allows Mbox to either send or receive real-time, full-framerate videos or stills between applications on the same machine. For example, it is possible to place the video output from Quartz Composer, Unity 3, or other applications on a layer within Mbox. Refer to "Using Syphon" on page 117 for more information about this feature.

TEXT FILES

Support for Text Files

Mbox can display text using RTF (Rich Text Format) or TXT (Text) files. Both files will work, but the RTF format will result in higher quality.

.rtf

An RTF file can be created by using the TextEdit application on the Mac computer. Simply type some text into a document and then save as an RTF file. Be sure to give the document a numerical prefix, make sure it has a file extension of .rtf, and place it in the *Media* folder.

Guidelines for RTF files:

+ Do not use multiple fonts on the same line of text.
+ Keep font sizes large.
+ Justification in the document is ignored; use spaces in the text instead.
+ Font color is taken into account. For best results, make text white rather than black.

.txt

A text file can also be created using the TextEdit application. In this case, save as a .txt file. Be sure to give the document a numerical prefix and place it in the *Media* folder.

Guidelines for TXT files:

+ Text quality is quite low in comparison to that of RTF files.
+ The .txt format does not allow for text sizing or custom fonts.
+ If there is more than one line of text in the file, Mbox will "play" the file line by line.
+ Adjusting play speed will cause the file to play faster or slower.

SCRIPTS

Using Scripts with Mbox

Beginning with Mbox v3.9, it is possible to trigger custom scripts. These scripts can be either AppleScripts or shell scripts. The two types of scripts are created differently, but both types can be triggered identically.

Scripts are numerically indexed using an 8-bit prefix, just like other content on the server. Scripts must be placed in the */Mbox/plugins/scripts* folder. AppleScripts must use the file extension '.scpt' and shell scripts must use the file extension '.sh'

<div align="center">001.Apple Script.scpt   or   123.Shell Script.sh</div>

Scripts are considered metadata within the Mbox Content Management system. Beginning with Mbox v3.9, metadata files can be sync'ed along with content (texture and object) files.

IMPORTANT! Scripts must be created outside Mbox using another application. They should always be tested outside Mbox first, and then tested within Mbox before use in show situations.

AppleScripts

Use the AppleScript Editor application to create and test AppleScripts.

AppleScripts can be used to control many aspects of the computer and other applications running on the computer. AppleScripts can also be used for external control, but require the script to trigger some other application for that external control.

CAUTION! In some cases, launching and performing tasks in some other application while Mbox is running, may negatively affect Mbox performance.

The AppleScript editor saves script files with the. scpt extension by default. You can also save the file as a text file (using either Mac or Unix line endings), but you must then change the file extension to .scpt manually before using the script in Mbox.

Shell Scripts

For simple scripts (single commands) use the Terminal application to test the commands and syntax first, then copy and paste the text into a text editor. For more complex scripts, the commands must be created in the text editor first and then tested.

IMPORTANT! Shell scripts must be converted to be executable before being used in Mbox.

To save a script as a text file and make it an executable shell script, do the following:

Step   1. Save the script as a file using the file extension '.sh'

Step   2. Open a Terminal window and type "chmod 755 " (make sure to include the trailing space) and then drag and drop the saved .sh file onto the Terminal window.

Step   3. Press [return] on the keyboard. (This step makes the file executable.)

Step   4. Double-click on the file to run the script for testing.

Shell scripts can be used to control many aspects on the computer, including external network communications. Shell scripts do not require other applications to be launched on the computer and therefore may have less impact on the performance of Mbox. However, shell scripts can be dangerous if they modify the core functionality of the computer or its communications.

More Information

There are many sources for information about AppleScripting and shell scripting on the internet and in book form.

Suggestions:

https://developer.apple.com/library/mac/documentation/AppleScript/conceptual/AppleScriptLangGuide/introduction/ASLR_intro.html

http://macosxautomation.com/applescript/index.html

http://en.wikipedia.org/wiki/AppleScript

http://macscripter.net/

AUDIO

Audio Output Features

Overview

In addition to movies, still images, and 3D objects, Mbox can also play audio files. Audio can be played back by any of the following methods:

+ The Mac computer's internal speaker.
+ The Mac computer's line-out or digital audio connectors.

Selecting an Output Device

In order for sound to play, the appropriate audio output device must be selected in the Mac's Sound Preference window. The balance and volume may also be adjusted at the Sound window.
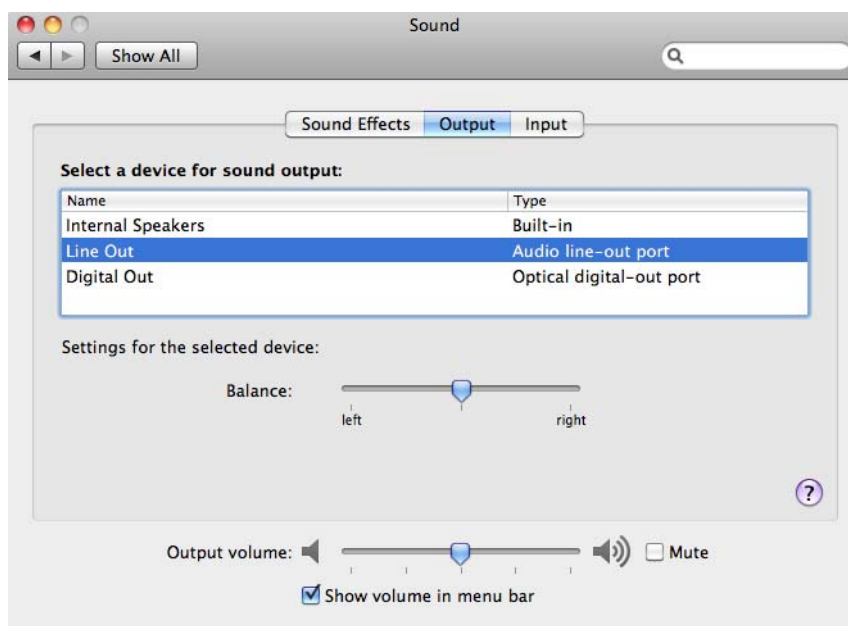
External Audio Tracks (Associated AIFF Files)

AIFF audio files can be placed in the *Media* folder with texture files. They are triggered when a content file with the same folder and file number is played.

Guidelines:

+ To play any associated AIFF audio files, the AIFF Playback checkbox must be checked on Mbox Setup tab (refer to "Setup: Options" on page 42). Remember that for this preference to take effect, the Mbox Mini software must be restarted after this change is made.
+ To trigger an audio file, it must have an identically numbered movie or still image and be placed in the same folder.
+ Associated audio files will only play in the forward direction and their play speed cannot be modified. Only the Forward Loop or Forward Once play modes can be used. If pause or scrub movie playback is used, the audio will not follow.
+ Audio files must be AIFF files - Stereo (L&R), 16-bit Integer, Linear PCM encoding, Big-Endian or Little-Endian.

+ The sample rate of the audio file must match the default sample rate of the computer for it to play at the correct speed (44.1kHz) .

+ A triggered audio file will always loop when the triggering movie loops (at the movie's out-point). If the audio is triggered by a still image, it will play once and not loop.

+ If an audio file is triggered with a movie, and that movie's playmode set to Forward Loop, the audio file will loop at the exact same time as the movie. This means that audio files that are longer or shorter than the triggering movie will not loop at the end point of the audio file. For an audio clip loop properly, it must have a movie of exactly the same length, down to the frame. Oftentimes you may be extracting an audio file directly from the QuickTime movie file that it goes with. If you do this properly, then your audio file will be exactly the same length as the movie.

+ Rather than having its own in and out points, AIFF playback uses those of the associated movie. If the in and out points of the associated movie are modified, the audio file's in and out points will be modified as well. If the play speed of a movie that triggers an audio file is modified, the audio file will continue to play at normal speed. This means that the audio will be out of sync with the movie and that it will loop too early (if the play speed is increased) or too late (if play speed is decreased).

+ An audio file can be triggered using a still image rather than a movie. Though in this case, the audio file will not loop. In this case, request a different content file on the same layer and then return to the initial triggering image to cause the audio file to play again. To play an audio file, but not see the content that is triggering it, simply request the triggering content with an opacity of 0. The audio file will play, but the triggering content will not be displayed.

Embedded Audio Tracks (in QuickTime Movies)

Mbox can also play audio that is embedded in a movie clip. To get Mbox to play the embedded audio in these files, it will be necessary to rename the file:

a file named - **001.MyMovie.mov** would be renamed as - **001.MyMovie.audio.mov**

In order to playback the embedded audio tracks, Mbox must play the movie in "Main Thread Mode." Main Thread Mode is the mode that allows playback of some less-desirable codecs and embedded audio. When playing audio in this manner, it has the advantage that the movie and audio playback are always synced, the playspeed can be adjusted, and the movie can be scrubbed. But Main Thread Mode also has drawbacks.

Guidelines:

+ It is not necessary to check the AIFF Playback checkbox (on the Mbox Setup tab) when using embedded audio.

+ Embedded audio tracks can use any audio format and rate.

+ Embedded tracks can also use as many channels as can be handled by the computer. Please note, however, that any audio output using more than two channels will require the use of the FireWire ports on the Mac computer and a compatible FireWire audio interface device (MOTU, RME, MH-Labs). (The Audio Playback checkbox on the Setup tab need not be checked for this feature to work.)

Audio Volume Control

A master volume control is provided by the Master fixture. (Refer to)

SYNCHRONIZATION

Timecode Synchronization (Timecode Sync Playmode)

Mbox allows movie playback to be synchronized to external SMPTE or MIDI timecode, or a simple internal clock. The timecode sync works with both 30 fps non-drop and 30 fps drop-frame timecode and requires the use of either the TC Reader application (explained later in this section) or the Mbox Remote application (refer to the Mbox Remote User manual).

By default, with no action taken by the user, every movie file is automatically linked to a time of 01:00:00:00 (HH:MM:SS:FF). Therefore, to simply trigger any file, set it to the timecode sync playmode (see below) and then run the clock signal up to 1 hour. The default timecode setting can be changed using the preferences available in the Mbox Remote application. The preference is called "TC Default" and the format is HH:MM:SS:FF. (Refer to the Mbox Remote User manual for more information.)

There are two timecode linking methods:

1) Using a .plist file to link files to timecode start times.

2) Using TC DMX Controls.

Guidelines:

  + Both timecode linking methods can be used at the same time.

  + Sync times generated by the TC DMX Controls method will override times generated by entries in the plist file.

Linking Files to Timecode (Method 1)

A Property List (plist) document can be created to hold the timecode values for any or all files on the server. The file should be called *MediaTimeCodes.plist* and should be placed in the *Mbox/Media* folder. This plist file is a simple set of Key:Value entries, where each key is a media index (folder.file) with a linked timecode value (HH:MM:SS:FF). Refer to the example window below:
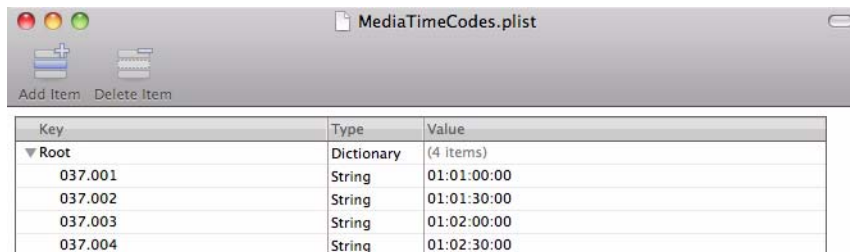
To create a Media Timecodes file:

Note: You will need to have an application that is able to create and edit Apple property list files. e.g. Property List Editor or PList Edit Pro.

Step   1. Open your property list editing application and create a new file.

Step   2. At window, make sure the root item is selected, then click **Add Child** button. At Key field, enter a number which is *library.file* number of the media file. Be sure to include leading zeros. You must use a period (.) as the separator between folder and file numbers.

Step   3. Press [tab].

Step   4. At Value field, enter a number which is the timecode value. You must use a colon as the separator between Hours, Minutes, Seconds, and Frames. Be sure to include leading zeros so that each element is two characters.

Step   5. Press [return].

Key	Type	Value
▼ Root	Dictionary	(4 items)
037.001	String	01:01:00:00

Step   6. Continue adding entries by pressing **Add Item** button.

Step   7. At File menu, select Save As. Navigate to the *Mbox/Media* folder, enter *MediaTimeCodes.plist* as the file name and click Save.

IMPORTANT! All values entered in the MediaTimeCodes.plist file <u>must</u> have the padding zeros. Folder and file numbers must have three digits, and the timecode values must have two digits. Incorrectly formatted values will result in those entries being ignored.

TC DMX Controls

In addition to assigning timecode values to individual media elements using the .plist method (refer to previous page), the timecode start value for a specific layer can be modified by patching additional control channels on any control universe.

Note: The TC Default preference also assigns a default timecode start time to all movie files even in the absence of other timecode start time settings.

The TC DMX Control feature allows patching of four control channels per layer to make live changes to the timecode parameters (HH:MM:SS:FF). This functionality is enabled by adding items to the "TC DMX Controls" preference in the Mbox Remote application, then patching the necessary channels on the control console. The control console may then be used to instantly set the timecode start time on the corresponding layer. (Refer to the Mbox Remote User manual for more information.)

To configure:

Step   1. Using Remote, open Mbox Server Preferences window (for the desired server).

Step   2. At "TC DMX Controls," click the [+] button to add a layer item to the parameter item list. (The item number relates to the layer number that the control will modify the timecode start time for.)

Step   3. In the text area to the right of the item number, enter the setting using the following format: <Universe>,<Address>. For example, the entry below would add a Timecode DMX Control for Layer 1 on Universe 101 starting at channel 1.
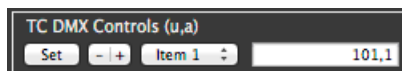
Step   4. Once the configuration is complete, click the [Set] button to accept the change. (Note: You cannot skip layers, so if you need timecode controls for Layers 1 and 3 you must also add a control for Layer 2, though it need not have a valid universe and address.)

Step   5. Click the [+] button to add additional layers as needed. (Added controls or edits to existing controls are immediately available for use without quitting and relaunching the Mbox software.)

To use a Timecode DMX Control, set each of the four channels to the desired values (8-bit numerical). For example, to use the patched channels from the example above, set Layer 1 to start at 02:15:25:02 and set the following values:

Universe.Address	Value
101,1	002
101,2	015
101,3	025
101,4	002

Guidelines:

+ Using the TC DMX Control with a movie file will override the TC Default and MediaTimeCodes.plists methods of setting timecode start time for a file.

+ Setting a specific item in the TC DMX Control list to a blank or invalid <Universe>,<Address> entry will disable TC DMX control for that layer.

+ Setting any of the four control channels to an invalid entry (e.g. setting Hours to 255) will disable the TC DMX Control and fall back to the MediaTimeCodes.plist or default TC value (in that order).

Using Timecode Sync Playmodes

Timecode Sync does not control any properties of a layer except the playback position and rate. All other functions must be programmed separately. If a movie is set to one of the three Timecode Sync playmodes, it will sit, paused on the in-frame until the appropriate time is reached. Thereafter, the movie will play based on the selected Timecode Sync playmode's characteristics until it reaches the out-frame, upon which it stops on that frame. It will not loop or fade out.

There are three Timecode Sync playmodes, each of which has its own benefits:

+ **130 - 134** TC Sync -> strict lock to timecode; movie waits for appropriate timecode to initiate playback, and if timecode stops or goes away, the movie playback stops.

+ **135** - TC Sync then Freewheel -> movie waits for appropriate timecode to initiate playback, but then ignores timecode for the remainder of that movie's playback

+ **136** - TC Jam Sync -> movie waits for appropriate timecode to initiate playback, if timecode stops or goes away, the movie freewheels. However, if timecode returns, then the movie will resync to the incoming timecode.

If the in-point of the movie is modified, playback will start from that point when the correct time is reached. The Timecode Offset channel on each layer allows modification of a file's start time by approximately 4 seconds without having to rename the file and rescan the content. The offset value options are:

+ **0 > 126** - Positive offset (movie plays later) -> 1 point = 1 frame

+ **127** - Default, no offset

+ **128 > 255** - Negative offset (movie plays sooner) -> 1 point = 1 frame

---

For a complete channel map, refer to

Receiving SMPTE Timecode (LTC)

SMPTE timecode input can be connected to the audio line-in port on the Mac computer. Both balanced and unbalanced SMPTE signals are accepted, as are both 30fps non-drop-frame and 30fps drop-frame formats.

"Line In" must be selected as the audio input source in the Sound preference window in the Mac's system preferences. The input volume can also be adjusted here.
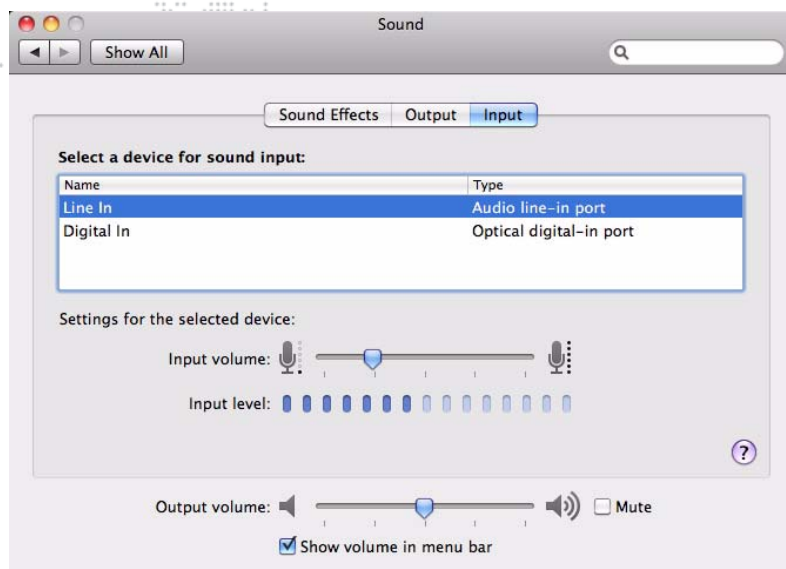
Receiving MIDI Timecode

A USB->MIDI interface box is required to receive MIDI timecode within Mbox Mini. (When using third-party devices, keep in mind that special drivers may be required for your computer.)

Setup for Timecode Input

The TC Reader application is used to set the preferences for incoming timecode. TC Reader contains a Timecode Reader window for this purpose. The preferences include the source type, output options, auto-start preference, and start/end time values. The Timecode Reader window also provides buttons for start, pause and resume control.

To toggle the Timecode Reader window on and off, press [⌘ T] on the keyboard.
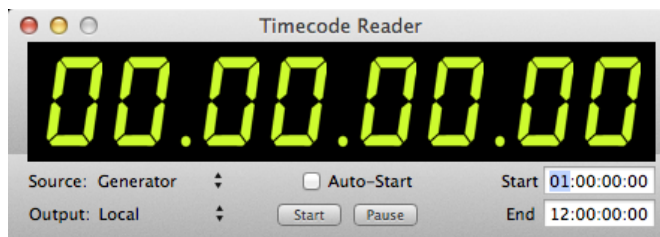
+ **Source** – choose the appropriate source for timecode: Generator, Audio In Left, Audio In Right, or MTC (MIDI timecode).

+ **Output** – Local (internal routing only), or an active Ethernet port (for sending timecode over the network to Mbox servers).

+ **Auto-Start** – enabling this will cause timecode be active as soon as the Remote application launches. Depending on the source setting, the application will immediately begin generating or listening for external timecode.

+ **Start** – starts generating code from Start Time value if generator is selected, or will start listening for external code if not.

+ **Pause** – pauses/resumes generating internal code or listening for external code (toggle action).

+ **Start** – specifies a time where the internal generator will start.

+ **End** – specifies a time where the internal generator will loop back to the start time.

Note: The Timecode Reader window does not have to be visible to function. If timecode is being received, it will be shown at the top-right of the Remote main window.

Guidelines:

+ Since TC Reader can retransmit the timecode data via Ethernet, a timecode source can be connected to one Mbox server and then retransmitted to any other networked servers. The servers must be networked together using a compatible network setup.

+ When using a timecode input, multiple layers on one Mbox server or layers on multiple servers can respond to the same timecode, but need not have any other layer properties set the same (except the Timecode Sync playmode, of course).

+ For receiving timecode for show purposes, the use of TC Reader is recommended over Remote. You should run TC Reader on each Mbox that needs to receive timecode and then connect the timecode signal to each Mbox separately.

CAUTION! The TC Reader application will display the perceived timecode variety at the top of the window. When timecode is being received, a number in parentheses will be shown to the left of the Auto-Start checkbox. (See sample screen below.) If this number is displayed in red and does not match the number shown at the top of the window, then the incoming timecode is not formatted correctly. Mbox can compensate for 29.97 fps non-dropframe timecode by enabling a preference in Mbox Remote. (Refer to the Mbox Remote User manual for more information.)
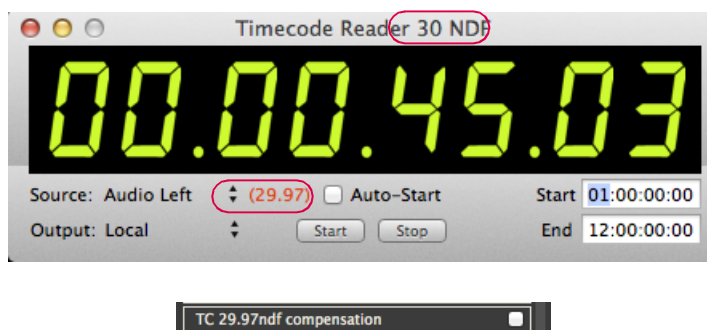
Layer-To-Layer Synchronization

Mbox offers the ability to synchronize playback of a designated master layer on one server to a layer on another server. A layer on one server can sync to the same numbered layer on another server, or one of two sync streams can be selected for both sync mastering and slaving.

Layer-To-Layer sync typically requires that most or all of the layer playback settings between two synced layers would be identical with exception of the playmode, which would designate master or slave:

+ **140 > 149** - Layer to Layer Slave
+ **150 > 159** - Layer to Layer Master - FWD Loop
+ **160 > 169** - Layer to Layer Master - FWD Once

In practice, it is possible to modify the in and out points of the layers, and to change the movie. This can lead to odd results, however, if the lengths of the resulting clips are not the same. Also, other settings on the server (Tile modes, edge-blending, etc.) can be modified. If you initiate a crossfade (at the same time) with both layers, the outgoing and incoming images will stay in sync.

To select which stream is used by a server while using Layer-To-Layer sync, adjust the Sync Stream channel appropriately on both Master and Slave as follows:

+ **0** - Layer to Layer by layer number: If Layer 1 on a server is set to one of the Master playmodes and Stream 0, then if Layer 1 on any other server (but not the same server) is set to the Slave playmode and Stream 0, the slave will sync to the master layer.

+ **1** - Stream 1: Any layer can be set to Master and Stream 1 or 2, and any layer on any server set to Slave and the same Stream # will sync to that master layer.

+ **2** - Stream 2

The Timecode Offset channel of the slave layer can be used to fine tune the sync by up to 127 frames in a forward or backward direction.

📄 For a complete channel map, refer to "Play Mode" on page 137.

Specifying the Network Port for Layer-To-Layer Sync

When using Layer-To-Layer sync, each Mbox server must be told which Ethernet port to use for synchronization. The Network section of the Mbox Setup tab can be used to select one of three Sync ports: Local, Ethernet 1, or Ethernet 2. Only the currently connected ports will be shown in the pop-up menu, so it may not have three choices in all cases.

To prevent the sync signal from transmitting from a particular server, select "Local."
Otherwise, select which of the Ethernet ports to use for the sync signal. In typical setups, the port should be the same on every machine, whether the server is sending or receiving sync. Ethernet ports used for Layer-To-Layer sync on all Mbox servers must be configured using compatible network setups.

Also refer to "Setup: Network" on page 41.
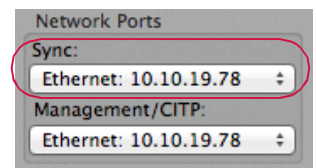
ALIGNMENT RECTANGLES

Using Alignment Rectangles

Oftentimes there is a need for program content to conform to a particular area of the output. This is especially true when outputting to a device that will crop the video signal or when using one server to drive multiple LED screens - using one layer per screen. In this case, Alignment Rectangles can be used to aid in the placement of content on the Mbox video outputs. Alignment Rectangles are drawn on the screen as the outline and can optionally display a crosshair and circle inside the rectangle. The crosshair and circle can be useful for assistance with overlapping and to check for proper aspect ratio.

A HUD that allows you to preview the location of Alignment Rectangles can be turned on and off by using the keyboard (see page 44) or by using the Master control channel (see page 135).

There are two modes of display for Alignment Rectangles: Rectangle Only and Rectangle with Crosshair and Circle.
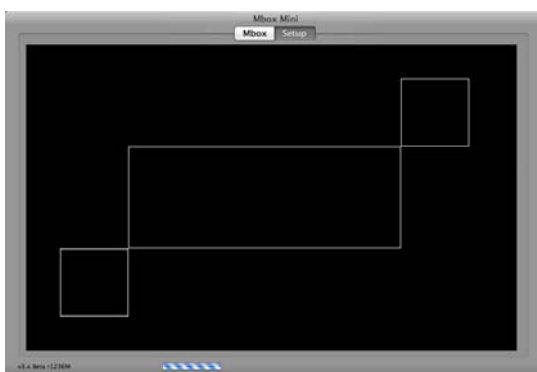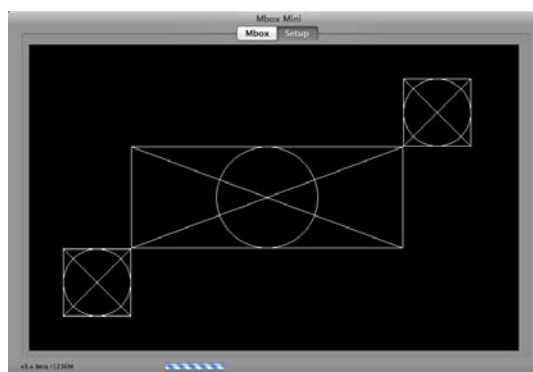
| Rectangle Only | Rectangle with Crosshair and Circle |

Alignment Rectangles have five settings; four are unique to each rectangle and one is global to all simple rectangles.

Every rectangle has and Origin X, Origin Y, Width, and Height setting. All rectangles also have a common lineweight setting and a custom color. Alignment Rectangles can be created by using the Mbox Remote application. (Refer to the Mbox Remote User manual for more information.)

Like many other settings for Mbox, the Alignment Rectangle settings are stored in a .plist file. This file's default location is */Mbox/alignment/000.alignment.plist*. As of Mbox v3.9, it is possible to have more than one alignment file (each prefixed with a unique index number) stored in this folder. Either the Mbox Remote application's server prefs window or the Master Control channel may be used to switch between alignment files on the fly.

It is possible to automatically crop and center content to any Alignment Rectangle. This is accomplished through the use of the effect called "AutoCrop."

#	Name	Description	Mod A	Mod B
212	Crop Content to Alignment Rectangle	Crops content to Alignment Rectangle	Rectangle #, 0-255	Value > 64 = Center content

If you choose not to center the content in the Alignment Rectangle, then only the portion of the content that overlaps the position of the Alignment Rectangle will be shown.

When using Panoramic Wide mode, both outputs together are treated as one rendering surface. Therefore, the origin of an Alignment Rectangle that appears on the right-hand screen needs to be offset by the width of the left-hand screen.

These features rely on additional information stored in the alignment.plist file. The Mbox Remote application is able to create these additional properties and edit them after they have been created. The supplemental application called "Mbox Alignment Rectangle Editor" can also help create and edit these more complex Alignment Rectangles.

Alignment Rectangle Grouping:

It is possible to have nine (9) groups of Alignment Rectangles. The group number comes into play when the Master control channel is set to a value that makes Alignment Rectangles visible. Rectangles can be assigned to groups using the Mbox Remote application. (Refer to the Mbox Remote User manual for more information.)

Master control channel values are as follows:

+ **120** - All rectangles without X & O
+ **121** - Group 1 rectangles without X & O
+ **122** - Group 2 rectangles without X & O
+ **123** - Group 3 rectangles without X & O
+ **124** - Group 4 rectangles without X & O
+ **125** - Group 5 rectangles without X & O
+ **126** - Group 6 rectangles without X & O
+ **127** - Group 7 rectangles without X & O
+ **128** - Group 8 rectangles without X & O
+ **129** - Group 9 rectangles without X & O

+ **130** - All rectangles with X & O
+ **131** - Group 1 rectangles with X & O
+ **132** - Group 2 rectangles with X & O
+ **133** - Group 3 rectangles with X & O
+ **134** - Group 4 rectangles with X & O
+ **135** - Group 5 rectangles with X & O
+ **136** - Group 6 rectangles with X & O
+ **137** - Group 7 rectangles with X & O
+ **138** - Group 8 rectangles with X & O
+ **139** - Group 9 rectangles with X & O

EFFECTS AND TRANSITIONS

Effects and Transitions with Additional Input Files

Effects and transitions that require additional image input files will use numbered files placed in a specific location inside the *Mbox* folder. The actual parent folder of the image files will depend on the effect or transition in question:

+ Alpha Mask From File Effect - *Mbox/plugins/images/masks/…*

+ Custom Wipe Transition and Shaded Material Effect - *Mbox/plugins/images/grayscale/…*
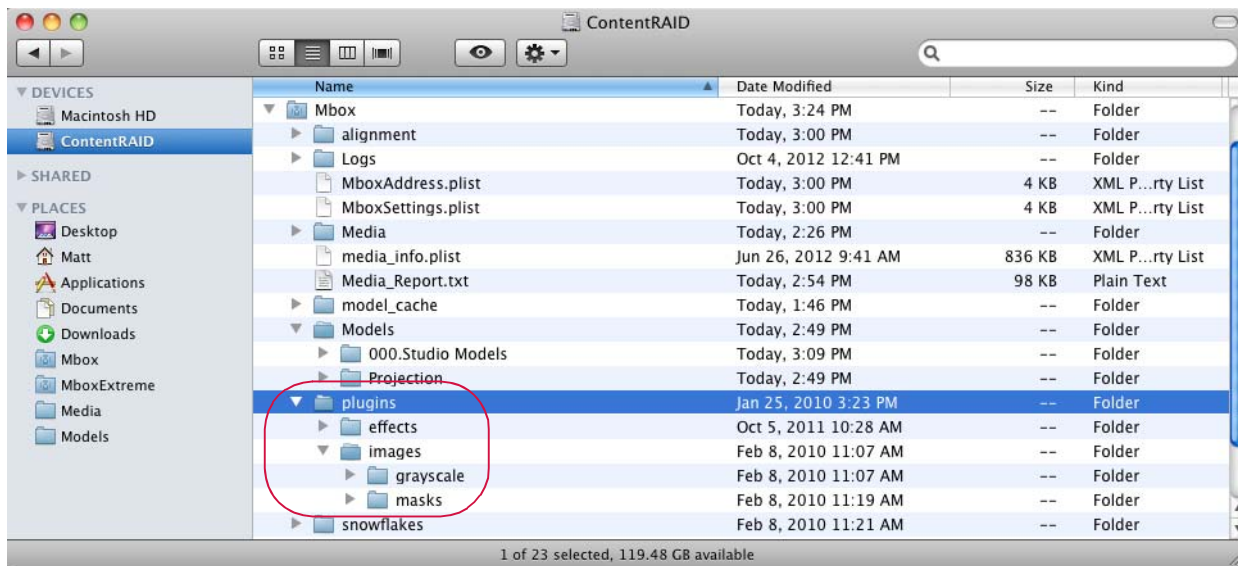
Within each of these folders, you may have numbered files. In the case of the Alpha Mask effect, you may have up to 255 files, numbered 001 – 255. In the case of the Custom Wipe transition, you may have up to 20 files – ten are hard-edge wipes 001 – 010, and ten are soft-edge wipes 011 – 020. Numbering, naming, and file extensions conventions follow the same rules as standard Mbox content. [e.g. 001.MyFile.png]

Guidelines:

+ For the best results, the input image files should be sized appropriately for their intended purpose. However, this is not necessary as the Alpha Mask effect and Custom Wipe transition input images will be scaled to fit the content they are used with. This can result in a mismatch between the aspect ratio of the input image and the content though. When using the **Custom Wipe**, for best results make sure the outgoing and incoming content are the same size.

+ The **Alpha Mask** input image file must be either a PNG or TIFF file saved with an alpha channel. The effect uses the alpha channel information in the input file to generate the mask.

+ The **Custom Wipe** input image file can be a JPG, PNG, or TIFF and can be grayscale or color. No alpha channel is required since that information is ignored. The grayscale file will provide the best indication of how the transition will work, as this effect uses luminance values to create the wipe. The darker areas will start to show the incoming content first, followed by lighter areas.

Training Video: Mbox Media Server Training Video Chapter 13: Transitions

Training Video: Mbox Media Server Training Video Chapter 17: Texture Effects

PROJECTION MAPPING

Overview

Projection Mapping is an advanced feature within the Mbox Remote application (separate from Mbox Mini). This feature, in its basic form, can used to configure output settings for simple keystone, warp, and multihead output.

For instructions on using Projection Mapping, refer to its section within the Mbox Remote User manual.

PIXEL MAPPING

Overview

Pixel Mapping is an advanced Mbox feature that allows mapping of specific functions of sACN, Color Kinetics KiNET, or Art-Net-controlled fixtures (RGB, Intensity, CMY, etc.) to each of the pixels in a composite video image. Pixel mapping data is generated from the onscreen imagery of the server and is output from an available Ethernet interface of the Mbox computer. While pixel mapping, Mbox retains the normal video output so a video signal can still be sent to a display device. Mbox also provides the ability to previsualize the pixel mapping setup on the video monitor.

Pixel Mapping for Mbox is set up using the Mbox Remote application. For instructions on using Pixel Mapping, refer to its section within the Mbox Remote User manual.

5.

PIXEL REMAPPING

This chapter provides instructions for the Mbox pixel remapping feature.

+ ABOUT PIXEL REMAPPING

+ PIXEL REMAPPING TUTORIAL

+ ADVANCED CONTROL

ABOUT PIXEL REMAPPING

Overview

The Pixel Remapping functionality of the Mbox software allows the creation of one or more two-dimensional rectangular objects for use on a layer. Each object can be textured by content used on the layer and can have properties that act upon its entirety or smaller regions thereof. Configuration options for Pixel Remapping allow for opacity changes, texture position changes, rotations, and object position changes. Further enhancements (not discussed in this document) allow for creation of repetitive identical elements with a regular offset.

Each Pixel Remapping file (stored at */Mbox/MultiScreenData.plist*) may have multiple configurations, and each configuration may be made up of multiple "screen objects" that can each have unique controls or mapping properties. External control over the screen object properties of the Pixel Remapping is handled through Art-Net.

Pixel Remapping is configured using the MultiScreen Editor application.
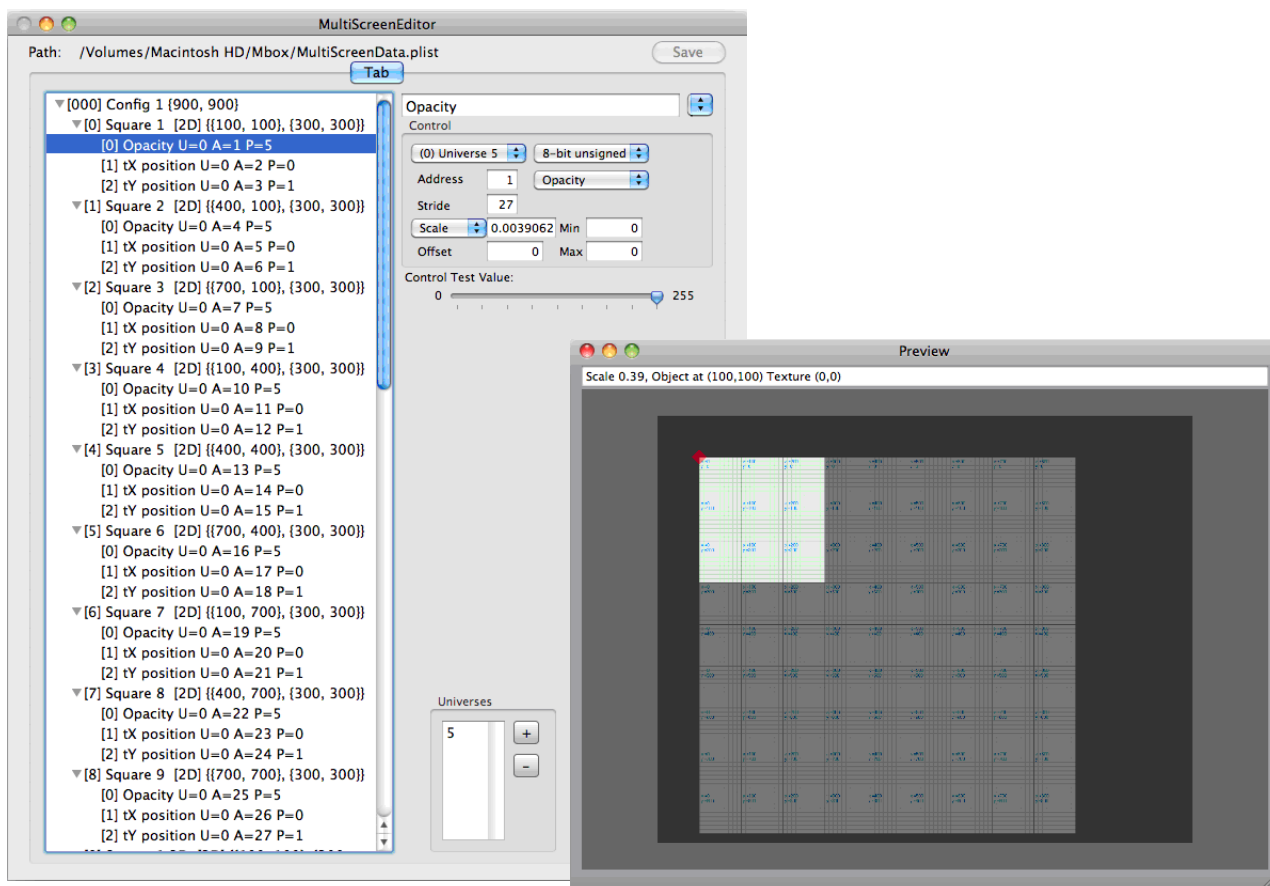
PIXEL REMAPPING TUTORIAL

 Training Videos: Pixel Remapping (9 videos): http://www.prg.com/mbox-training-videos/

Overview

The concept of Pixel Remapping can be best explained using a tutorial. The example provided in the following tutorial will use a square that will be broken into nine smaller squares. The following image represents this square:

The tutorial will create two configurations. One configuration will be set up as "Projected Mapping" - as if a projector were pointing at some screens that can move through the projector's beam. The second configuration will be "Discrete Mapping" - where the image on each square will stay pinned to square as it moves in space. To accomplish these modes, each of the smaller squares (the screen objects within each configuration) will need certain controllable properties - opacity, texture position, position - depending on the intent of the configuration.

External control over the screen object properties of the Pixel Remapping is handled through Art-Net. While this Art-Net can be patched to the same universe as Mbox, this is impractical unless very few channels are required. Ideally, a unique universe should be used. In the case of moving scenery, this external Art-Net may be derived from positional data that a scenic feedback system creates and which is then converted into Art-Net. In other cases (and in this tutorial), you can just patch extra fixtures/channels on a console to generate the necessary Art-Net.

For the purposes of this tutorial, each of the nine squares will need three (3) extra channels patched per layer to control its properties: X Position, Y Position, and Opacity. You need per-layer controls because the gobo can be used on each layer of Mbox and might require separate, unique control of some aspects on a per-layer basis. (Note that in the case of scenic tracking, per-layer position control may be unnecessary, but opacity may be necessary.) You'll use 8-bit control for everything in this tutorial. So with nine squares at three (3) channels each, and six (6) layers set up on the server, you will need to receive 9 x 3 x 6 = 162 additional channels from some Art-Net source. If you can write a profile for the console, these could be set up as a channel for Opacity, and then a pair of channels for X and Y Position. 54 of these three-channel fixtures could be patched (9 squares x 6 layers).

To make the math simpler, make the dimensions of the large square 900 pixels on each side, and the smaller squares 300 pixels on each side.

Projected Mapping

Pixel Remapping is configured using the MultiScreen Editor application. To open the MultiScreen Editor, locate its icon in the *MboxExtreme/Utilities* folder and double-click.

Upon opening the MultiScreen Editor application for the first time, you are presented with two windows and a blank workspace:

+ **MultiScreen Editor (main) window** - allows numerical data entry to set up the configuration and the screens.
+ **Preview window** - a scalable view of the selected configuration and screens.
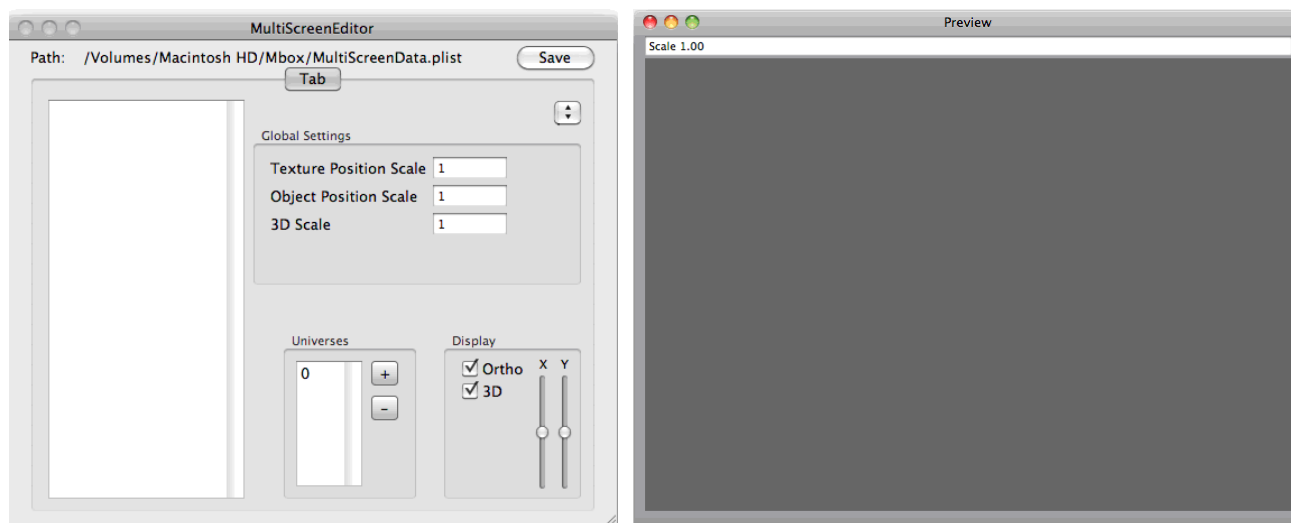
Note: The Mbox software can currently only use the file called MultiScreenData.plist which must be located in the *Mbox* folder. The MultiScreen Editor application can open, edit, and save configuration files with different names, but Mbox won't be able to open them unless they are manually renamed in the Finder.
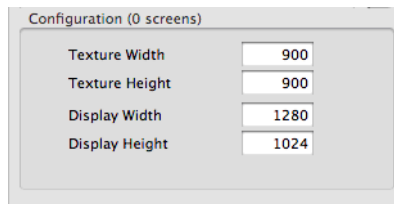
The first two steps are to set up the additional universe of control (via Art-Net), and to add a configuration.

Step 1. On the main window, double-click on the "0" in the Universes area - it is defaulted to 0 - and type in a new number (this tutorial uses universe 5) that is not one of the current control universes for your Mbox server. Press [return] to accept the change.

Step 2. Next, click on the small, blue pop-up handle at the top right of the window and select **Add Configuration** from the list. Don't change any of the Global Settings, or the settings in the Display area.
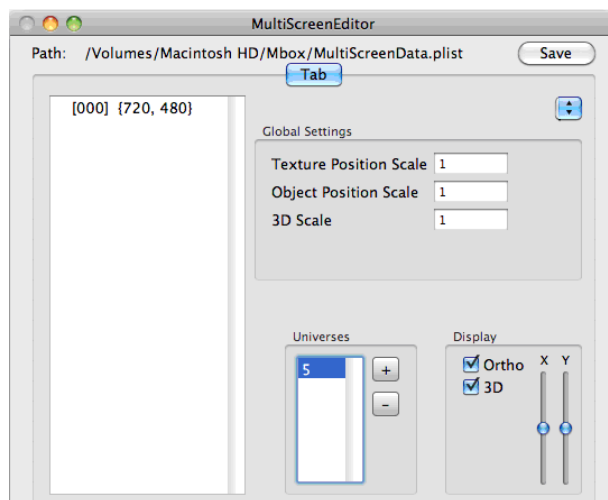
You can see the added the entry in the list (the one that says "[000] {720, 480}"), this is your first configuration (with index 0). You can have more than one configuration (by the end of this tutorial we will have two) with each having the same screen objects (same size/position of screens, etc.), but each configuration may have different pixel remapping, scaling, or additional controls. All configurations get saved into the same MultiScreenData.plist file and can be selected (using an index value between 0 and 255, which will correspond to the number of the configuration) at runtime using the Animation Modifier A parameter each layer. In this way, different layers can use different configurations from the same file. To start with we'll only add the one configuration.

Step   3. Click on this configuration ([000]) to select it in the listbox at the left of the window and you will notice that the window changes from showing the Global Settings to showing Configuration settings; specifically Texture Width & Height and Display Width & Height. Edit those values to be 900 x 900 and 1280 x 1024 respectively, make sure to press [tab] or [return] after each entry.

Configuration (0 screens)	
Texture Width	900
Texture Height	900
Display Width	1280
Display Height	1024

The Texture dimensions are typically the pixel size of the object and, in this example, happen to be the size of the matching test content because we have made content specifically for the object. The Display dimensions are the pixel size of Mbox's fullscreen output. You can use files with pixel dimensions that don't match the Texture Width & Height set here, but in this case you should test the configuration before show use. You can also use the configuration when the Mbox fullscreen output size is set to different values, but the object's location on the screen may not be in the right place.

Note: Content files with different sizes will be centered on your MultiScreen object and not offset from the top-left corner of the object and/or content. MultiScreen objects themselves are always offset from the top-left corner of Mbox's fullscreen output.

Step   4. For the purposes of this tutorial, 720 x 480 stock content will work with the configuration we are creating, perhaps not as well as content sized for the object. Before proceeding any further, type a name for the configuration into the empty text field at the top right and then press [return]. Note that the name you entered will appear in the list. Then click the save button.

Note: You can adjust the Editor's window size by dragging the bottom-right corner.
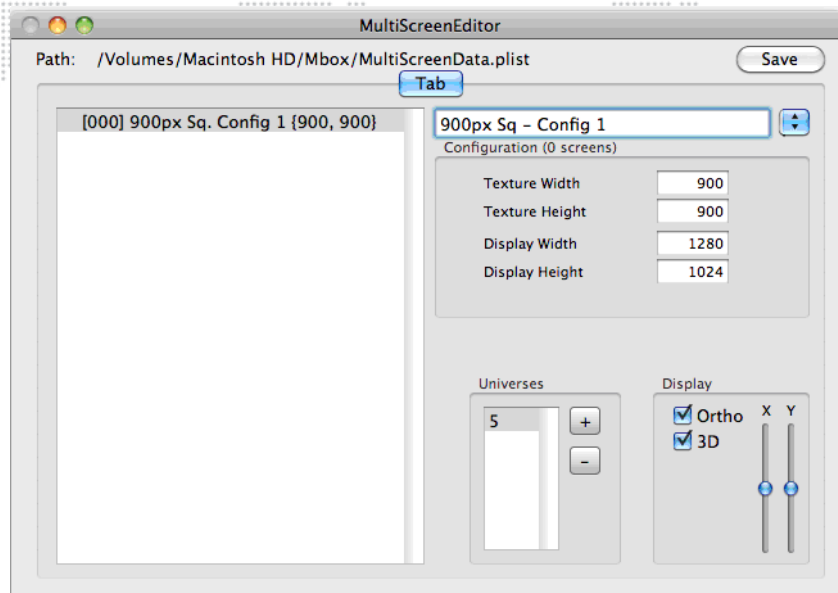
Step 5. Click on the small, blue pop-up handle again (with configuration [000] still selected) and choose **Add Screen** from the list. You'll see a small disclosure arrow appear to the left of the configuration's entry in the listbox. Click on this arrow to expand it. You'll see that the configuration now has one screen (Screen index [0]), you can select this screen and give it a name, and also see its properties on the right side of the window.
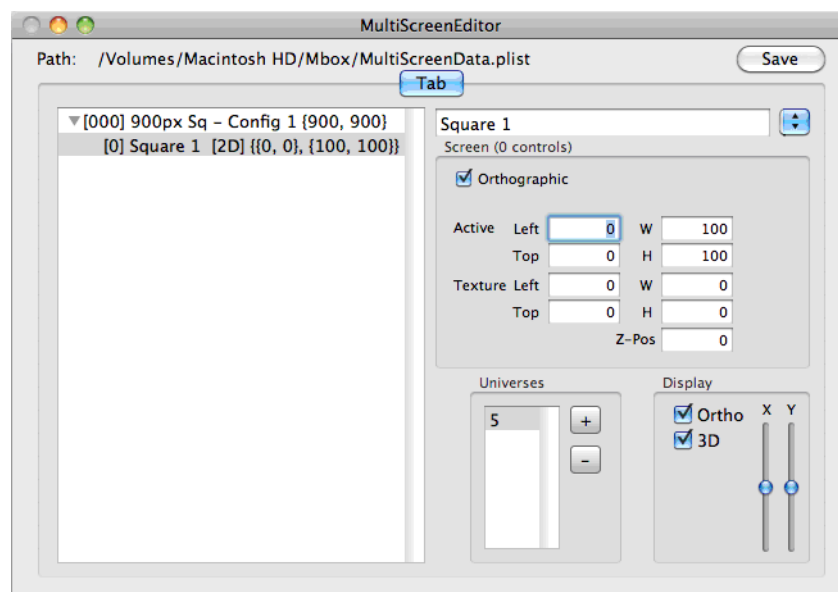
Step 6. With the first screen selected, take a look at the Preview window of the application. Initially it will look blank, with only a dark grey background. If you press [command -] on your keyboard you will reduce the scale of the view whereas pressing [command +] scales it up. You can also click and drag on the background to pan the view.

Note: The text field at the top of the Preview window shows the current scale of the view and information about the selected screen.

Step 7. Reduce the scale and center the background within the view. The selected screen appears at the top-left of the background, with the screen's top-left corner (the selected screen's origin) indicated by a red diamond. If you scale the view up and then pan up and left, you can see that the screen is represented as a piece of graph paper, with overlaid grid lines x/y coordinates. These coordinates refer to the intersection of major gridlines and not to the screen or entire MultiScreen object. They are shown for visual reference as to the size and location of the screen.
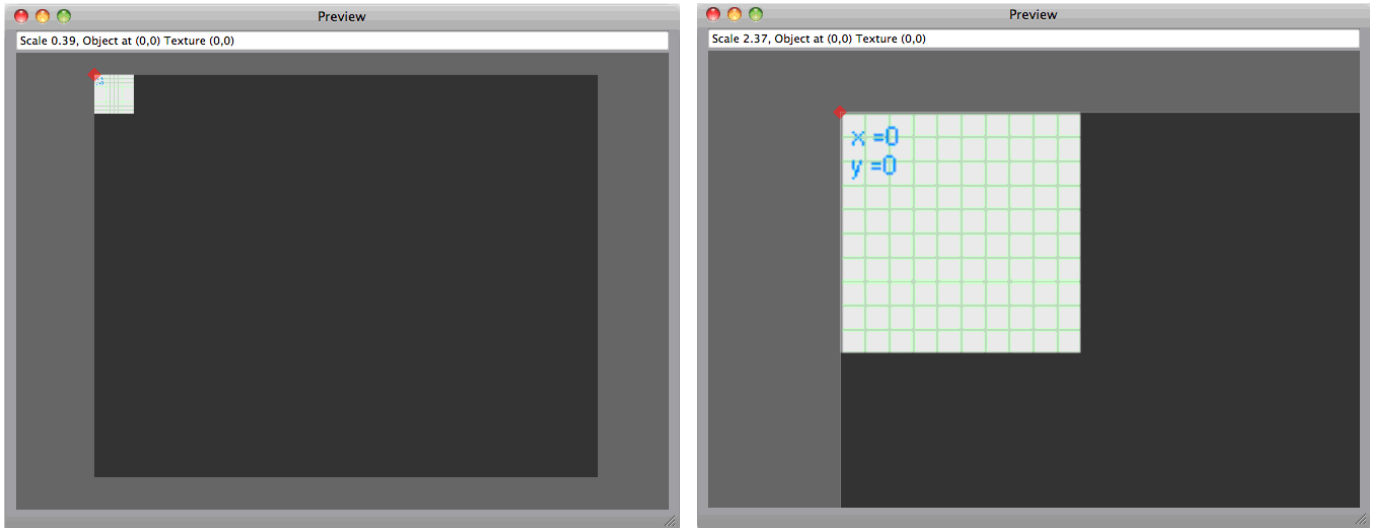
You now need to enter useful values for the screen properties. But first an explanation of each of the properties (all origins are based on the top-left of the Mbox output, the MultiScreen object, and the screen object being 0,0):

+ **Active**: Origin and dimensions (in pixels) of the rectangular area on Mbox's output screen where the selected screen (and its applied texture) will be drawn. In most cases, the Active Width & Height will be the same size as the Texture dimensions unless some scaling is desired while remapping the pixels.

+ **Texture**: Origin and dimensions (in pixels) of the rectangular area of the layer's texture that will be applied to the selected screen belonging to the current MultiScreen configuration. These numbers are generally very exact, especially with custom pixel-for-pixel content made specifically for LED screens.

CAUTION! To ensure that content is centered on MultiScreen objects by default, all Texture origin values are evaluated taking the current configuration's Texture Width & Height into account. This means that if you set the configuration's Texture Width & Height to a size that isn't the same as the content's size, then the top-left corner of the content won't appear at the top-left of the screen.

Using our example: with a texture that is exactly 900 x 900, this texture's top-left corner (its 0,0) can be placed onto the top-left corner of the screen object by using a Texture origin of 0,0 as well. But with a texture that is 720 x 480, the screen's Texture Left and Top values must be set to 90 and 210 respectively to get the top-left of the smaller content into the top-left of the screen object. This is because the smaller content has to be offset to the left and upwards to get its top-left corner into the top-left corner of the screen object. The math: Left = (configuration width - content width)/2, Top = (configuration height - content height)/2. If the content is larger than the configuration's texture size, then the offset values will be negative.

+ **Orthographic Checkbox**: This selects whether the screen is treated as 2D (box is checked) or 3D (not checked). For typical usage, the screen objects that you use for actual output to on-stage displays will be of the 2D variety. Unchecking the Orthographic checkbox allows you to create 3D screen objects for use in previsualization of your screen objects. With some minor configuration changes to Mbox, these 3D screen objects can be shown

alongside the 2D screen objects. For previsualization purposes, the viewing perspective of 3D screen objects can be altered by repositioning the Mbox camera.

+ **Z-Pos**: Used with 3D screen objects to adjust the distance of the screen object from the camera in order to correctly position multiple screens (to represent their position in the real world) or to ensure that screens that can overlap look correct when doing so. Unlike the Active and Texture values, the Z-Pos value is not scaled in pixels. (Some experimentation may be required to get the right value. Values may be positive or negative, with positive values being towards the viewer and negative being away from the viewer.)

Step 8. For the nine square screen objects you are creating in this tutorial, make them 2D, and make the assumption that they are being used on LED panels that can be moved in space through some sort of scenic movement system.

    a. For the **Active** origin of the first screen object, think about where you would like it to appear on the screen. Where it should appear on the screen may be totally arbitrary, or may be driven by the device that the video output is being sent to - an LED processor perhaps - in which case someone else may be telling you exactly where the image should appear on the screen in order to be mapped correctly on the LED wall. For this example make the origin (Left, Top) of this first screen object 100,100. This individual screen object is 300 pixels on each side and you don't want to do any scaling on the screen, so the Active width and height of the square will be 300 and 300.

    b. For the **Texture** origin of this screen object, you should use 0,0. This is because you want to take the top-left ninth of the source texture (the PNG with the numbered red squares) and have is be applied to the rectangular area of the screen defined by the Active origin, width, and height. Your Texture dimensions will be 300 x 300, to match the portion (1/9th) of the larger red square that you want to apply.

The next step would be to add the remaining eight screen objects that make up the remainder of this MultiScreen configuration. But before you do that, think about a couple of other things. The ability to copy and paste within the MultiScreen Editor application makes it quick to create new configurations, screen objects, or controls once you've laid out the measurements and the properties of the first object.

For now, here's a chart with the Active and Texture origins/dimensions for the remaining eight screen objects in this first configuration. Later on these values can be applied to the respective screen objects.

Table 5-1: Active and Texture Origins/Dimensions

	Square 1	Square 2	Square 3	Square 4	Square 5	Square 6	Square 7	Square 8	Square 9
Active Left	100	400	700	100	400	700	100	400	700
Active Top	100	100	100	400	400	400	700	700	700
Active Width	300	300	300	300	300	300	300	300	300
Active Height	300	300	300	300	300	300	300	300	300
Texture Left	0	300	600	0	300	600	0	300	600
Texture Top	0	0	0	300	300	300	600	600	600
Texture Width	300	300	300	300	300	300	300	300	300
Texture Height	300	300	300	300	300	300	300	300	300

Step 9. Before adding the other eight screen objects, you're going to add some controls to the first screen object. Remember that we said you would need Opacity, X Position, and Y Position controls for each screen object. With the Square 1 screen object selected in the MultiScreen Editor app, click once again on the small popup handle at the top right of the window. This time select **Add Control** from the list. As when you added a screen object, a disclosure arrow will appear in the listbox. This time it will appear to the left of the screen object's name. Go ahead and click on it to expand it.

Step 10. You now have one control attached to the screen, give it the name "Opacity." As with the properties for the screen object, each control has properties that can be edited:

a. **Universe**: If this control is to use external Art-Net data, this is the universe that it will listen to. The list is populated with the pool of Art-Net universes provided by the listbox at the bottom right of the main window.

Note: The Universe popup shows a number in parentheses to the left of the absolute universe number. The number in parentheses is the zero-based index number of that universe within your group of universes used in the entire MultiScreen setup.

CAUTION! If you edit any universe values in the Universes listbox those edits will not be automatically updated to the universe value for controls that have already been configured. You must reselect the universe value for those controls.

b. **Address**: This is the address of the first channel within the selected universe that this control will listen to for its data. Values between 1 and 512 are valid.

c. **Stride**: This number represents the total number of Art-Net channels that this screen object uses (for all of its controls, not just this one). The stride setting simplifies the use of the MultiScreen object on more than one layer so you don't have to perform excess patching or create multiple configurations, one for each layer. The stride setting tells the screen object how many channels to skip between layers, so it can automatically use the correct address offset for each layer. In the case of some controls where you don't want each layer to have a different/unique value (e.g., positional data) the Stride value should remain at 0. For controls like opacity that might need a unique value for each screen for each layer, the stride setting becomes very useful.

Step 11. Because some controls need to have a per-layer value and some do not, this leaves several options on how to patch everything. You could give every control a unique address for control on every layer. This would require the use of more control channels and might cause you to have to send the same values to more than one control at the same time. One alternative is to use the stride setting, but to use the same address for controls that need to receive the same data. This cuts down on repeated data, but leaves empty channels in your patch. The other alternative is to patch all the unique control channels sequentially before controls that use a stride setting and then patch the per-layer controls.

a. **Data Format Pop-up**: This sets the range and default value of incoming Art-Net data. As with typical fixtures, 8-bit data uses one channel (0 - 255), 16-bit uses two (0 - 65535), and 32-bit signed uses four (-2,147,483,648 - 2,147,483,648).

1) In the case of the 8-bit controls, "unsigned" relates to a control with a range of 0-255 and a default that is one end of that range (usually 0), "128" relates to a control with the same range but a default of 128 and which varies above and below the default, and "127" is like "128" but with a default of 127.

2) In the case of the 16-bit controls, "unsigned" is as with the 8-bit version noted above, but with a range of 0-65535. "Offset" relates to a control like the 128/127 varieties of the 8-bit controls, with a default at halfway, i.e. 32767.

3) And finally, the 32-bit signed control has a very large range that is both negative and positive. 32-bit signed controls are generally reserved for positional feedback from very accurate scenic encoders as the large range allows the control to be extremely accurate, receiving actual encoder counts in many cases.

b. **Control Type Pop-up**: This is where the property of the screen that is to be controlled is selected - position, texture position, rotation, etc. Refer to "Screen Object Control Types (Units, Typical Scale Value)" on page 104 for more details on the control types.

c. **Scale/Counts Pop-up**: This switches between the two methods of scaling the incoming Art-Net data to apply it to a control.

1) For example, if you are receiving a 16-bit offset value for X Positioning, but only ever need to move a screen object 100 pixels left or right, then you may want to relate a range of 0-65535 to a range of negative 100 to positive 100. Since the total range of movement is 200 pixels, your Scale value would end up being 200/65535 = 0.0030518. Alternately since the movement on either side of the default value is 100 pixels, the scale could be calculated as 100/32767 = 0.00305185 .

2) Alternately, counts (typically used for linear position controls) equates to the number of Art-Net steps (counts) received per pixel of total travel. With the above example, since you are using 32767 steps to move the screen object 100 pixels, your Counts value ends up being 32767/100 = 327.67

3) The two methods are just the same information presented in a different ways, one is the reciprocal of the other: 100px/32767counts = 0.00305185 (scale = pixels per count) and 32767counts/100pixel = 327.67 (counts = counts per pixel).

Note: The sign (+ or -) of the scale value is be used to invert the property's direction of adjustment relative to the incoming data.

---

d. **Min and Max**: These values are typically used to limit the travel of a linear control used for scenic tracking to a specific number of pixels; so as to keep the screen object (2D or 3D) or its applied texture from moving too far if incoming tracking data values exceed the expected limits. Their use with non-linear controls may not be necessary, as the scale/counts value for non-linear controls tend to be set more precisely.

e. **Offset**: The Offset value is used to preset a value for a control that does not respond to external data. Offset is typically used with the Repeat controls, but can be used to move or rotate a screen object a fixed amount. When using an offset value the Scale value should be set to zero (0).

Step 12. The opacity control that you have added now needs to be configured to respond to some Art-Net and also to have the appropriate scale. First of all, use the universe popup to set the Universe to "(0) Universe 5", and leave the address at 1.

Step 13. Choose "8-bit unsigned" for the data format, and "Opacity" (of course) for the control type. You'll need to do some math for the stride value - assuming three channels per screen object (opacity, X position, & Y position), nine screen objects, 3 x 9 = 27. And finally for your scale, an 8-bit control has a range of 0-255, and the opacity parameter has a range of 0.0-1.0 so you need to divide 1 by 255 = 0.00392157. You don't need an offset, min, or max.
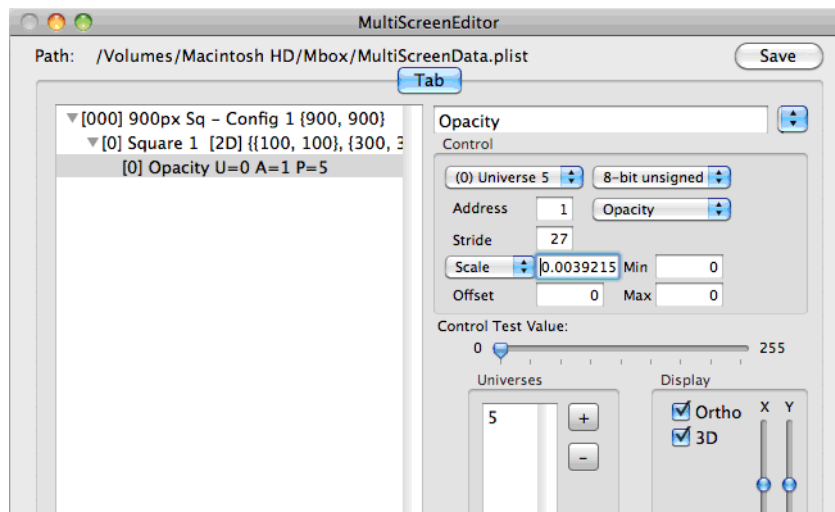
Step 14. After adding the opacity control press the **Save** button; then you can add the X and Y position controls. Here's where you can speed things up by using copy and paste. Make sure that the opacity control is highlighted in the listbox, press [⌘ C] and then [⌘ V], and you'll get a copy of the opacity control. Rename it "tX position."

    a. For the Texture X position control, the universe is the same, the address is 2, the data type is "8-bit 127", stride is still 27. Two complex decisions remain though. For the control type; do you want to use Texture X Pos or just X Position? And what should your scale value be? For this particular screen object in this configuration (using projected mapping) you want to select "Texture X Pos" because you only want to move the texture on the small square, not literally move the entire screen object on Mbox's output screen. You don't actually need to move the square in the 2D view - if you imagine an LED screen moving left and right on-stage, to keep projected mapping correct you only need to move the texture in the opposite direction to the screen's movement. If the screen object were to move on-screen, it could move away from the area that the LED processor is using for the LED screen.

Note: Why would you want to actually move the screen object? For your 3D representation (which will be covered later), you want to see the actual square move on the screen to mimic the movement of the LED screen on-stage. Also, if you were using this for a system with an actual projector instead of LED's, then you might need to move the screen object as the piece of scenery is moved.

    b. For the scale value, you want to be able to move the texture on this screen object so any part of the applied 900 x 900 texture can be seen. Since the screen object is 300 x 300, this means you only ever need to move the texture a maximum of 600 pixels in either direction (you may eventually want to set min and max values, but not now). With an "8-bit 127" data format, its range being 127 counts in either direction, you need to divide 600 pixels by 127 counts to get a scale of 4.72440945.

TIP   Regarding accuracy of scale values. The more accurate you can be with your scale/counts calculations, and the more decimal places you use when entering those values, the more accurate the screen object's responses to input will be. For linear movements, you should use at least 4 digits after the decimal point.

Step 15. Copy and paste the X Position control you just created, and rename it "tY Position." Change the address to 3, and the control type to Texture Y Pos. Press the **Save** button.

Note: Once you have one or more screen objects created in the editor, you'll want to test your configuration in Mbox to make sure you have the correct sign on the scale value for your X and Y position properties. In most cases, particularly with Art-Net-only control (i.e., no scenic encoders), once you figure out the signs for one screen object you can use the same signing convention on all following screen objects. For this tutorial, leave the texture X position's scale as positive, but make the texture Y position's scale negative.

There is a Control Test Value slider on the main window that changes scale based on the selected control's data format, and allows you to test linear controls (e.g., position, opacity) within the MultiScreen Editor application. If you look at the Preview window, the text field at the top will show you both the screen object's location and the texture location. As you move the slider, you should be able to see these values change and also check to see if the direction is correct.
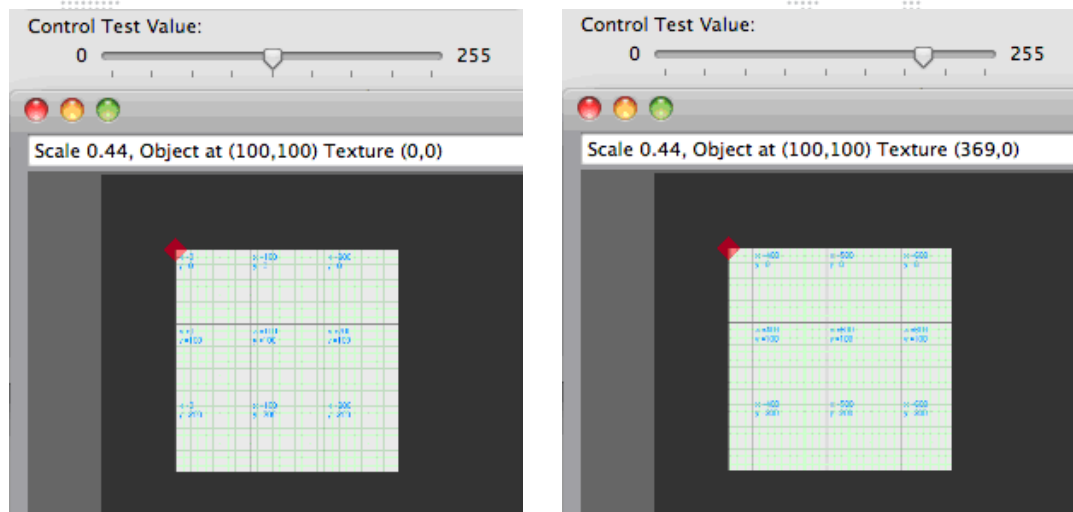
Step 16. Now you can go ahead and add eight more screen objects by copying and pasting the one screen object you have just configured. Make sure to use the correct Active and Texture origins and Width & Height from the chart on , and also to substitute the correct address for each control. You should end up with what you see below:
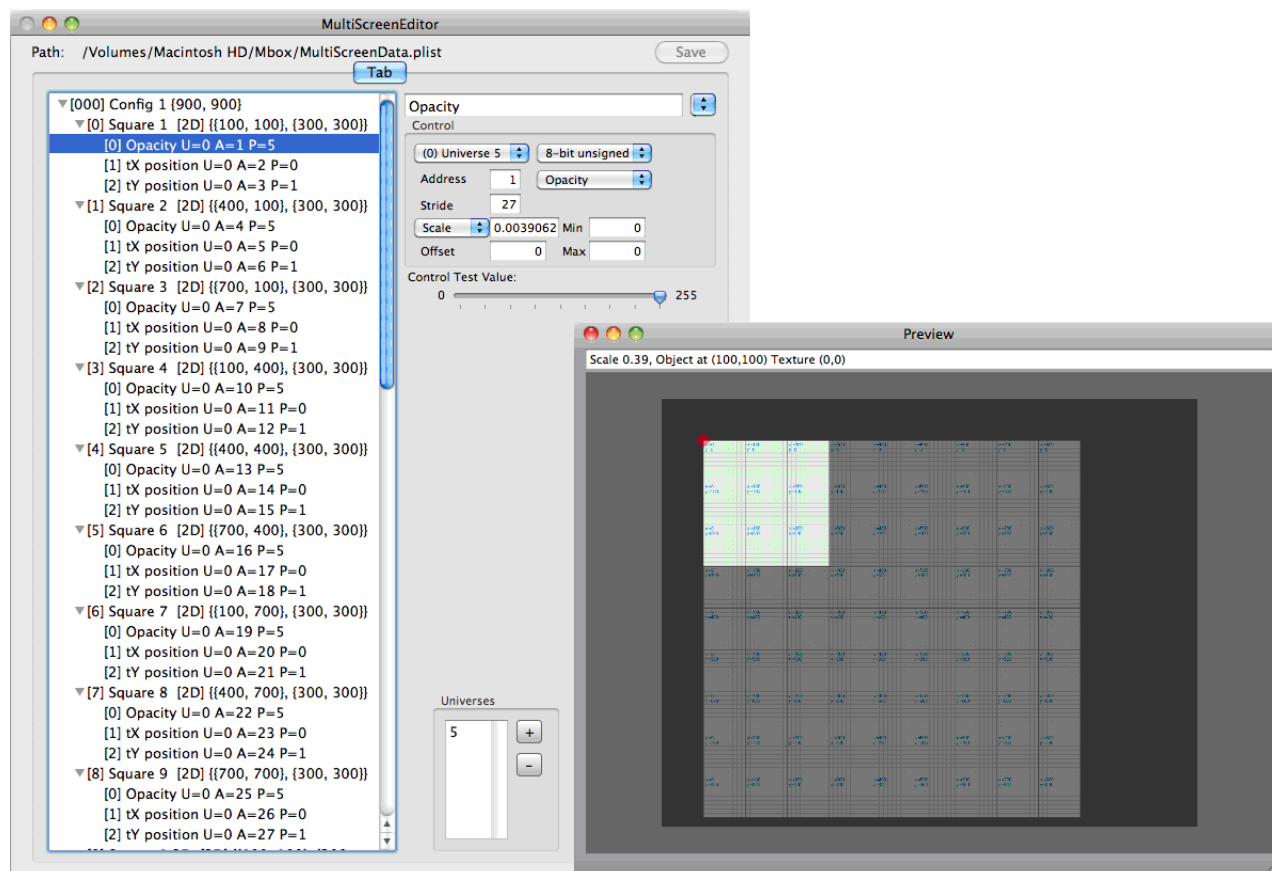
Step 17. To test the configuration you have created on Layer 1 - select Object folder 2, file 9 for the MultiScreen object, select the sample PNG as the texture on the same layer, and also supply the correct additional Art-Net data (for the opacity and texture X & Y position) on Universe 5. You should see something like this:
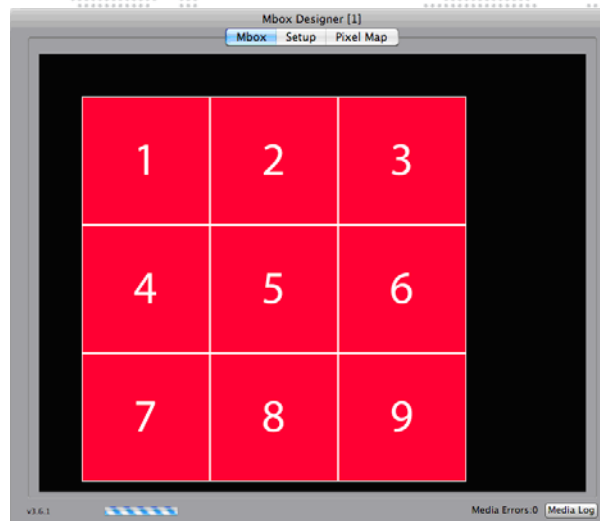
Note: Since the texture X & Y position controls default to 127, you will need to set them to that value to see the portion of the texture in the illustration above. With different values, the texture is going to move, and where it moves off the surface of a screen object, it becomes fully transparent.

Note: The Mbox Fullscreen mode resolution is set to 1280 x 1024 to match the MultiScreen configuration's Display Width and Height.

If you adjust the levels on universe 5 you will be able to adjust the opacity of each of the nine square and adjust the portion of the texture that is displayed on each square. You can also use the MultiScreen object on other layers by using the correct Art-Net offset to control it.

But what about the MultiScreen object's 3D visualization functionality? It's pretty simple to get the MultiScreen configuration to do this, and easy to get Mbox to display it. The idea, when using the 3D element of any configuration, is that a specific area of one output of the server (or the entire output) will show the 2D representation, and some other dedicated area (or the other output of the server) will show the 3D representation. The 2D output will generally be sent to the actual video output and to the stage, while the 3D output can be used for pre-visualization of the setup.

To best accomplish this, you will need to set Mbox into Panoramic Wide mode, and Mbox will need to be configured to force the 3D representation onto the second output. Because this functionality of the MultiScreen object is actually 3D, its position and your perspective view of it can be adjusted by manipulating the position and orientation of the standard camera - Orbit X & Y, Dolly, etc. But since the camera doesn't have lateral movement control that can be addressed via Art-Net, you will need to use Mbox Remote to move the camera such that the 3D representation moves onto the second output screen. (More on how to achieve that below.)

Step 18. The first step is simple, just select each screen object (Square 1, Square 2, etc.) in the configuration one at a time and copy and paste it into the same configuration. Rename these new screen objects (for example, "Square 1 3D") so you can tell them apart from the originals. Then uncheck the Orthographic checkbox for each of these new screen objects. The other thing that you will want to do (remember that you want to mimic LED panels that are moving and using projected mapping) is to add controls to the 3D screen objects for X & Y Position, in addition to the existing Texture X & Y Pos. Just use copy and paste on each of the original controls, and change the name control type for the newly created controls.

Step 19. On the main window of the MultiScreen Editor application, uncheck the box labeled "Ortho" in the Display section. You should end up with this:
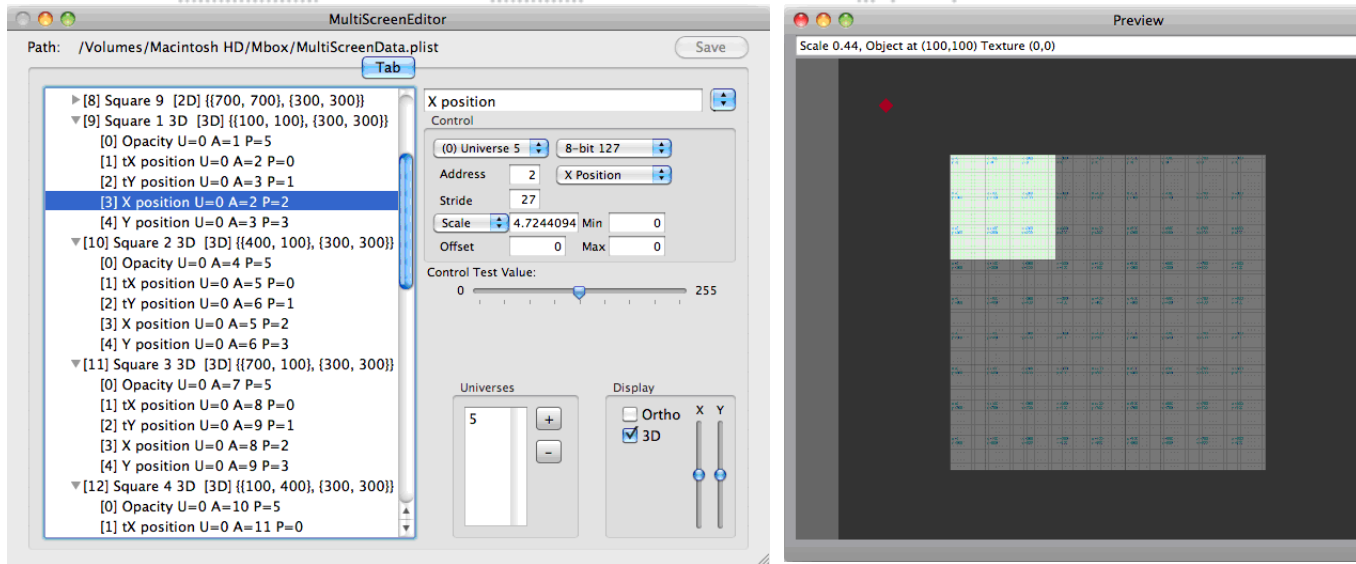
Step 20. You can use the Control Test Value Slider to mimic what will happen when this multiscreen object receives data. Because the Texture X Position and X Position controls (likewise the Texture Y and Y position) use the same address, when you move the slider, the 3D screen object moves and the texture moves on that object.

TIP In order for the 3D screen objects to move upwards when the control value increases, their scale needs to be negative. However, we previously set the Texture Y Position controls be negative as well. In order for the texture to move in the correct direction (you want them to move in the opposite direction to the movement of the 3D screen object), you'll need to change the sign of these controls' scale value back to be positive rather than negative. You will need to do this for the tY position control on all 18 screen objects in this configuration.

Step 21. Once you have everything correct, you can rearrange the screen objects, both 2D and 3D, using the Control Test Value slider.

a. Because this is a projected mapping configuration, the texture (the graph paper) should always stay in the same place in the 3D representation no matter where the 3D screen object is.
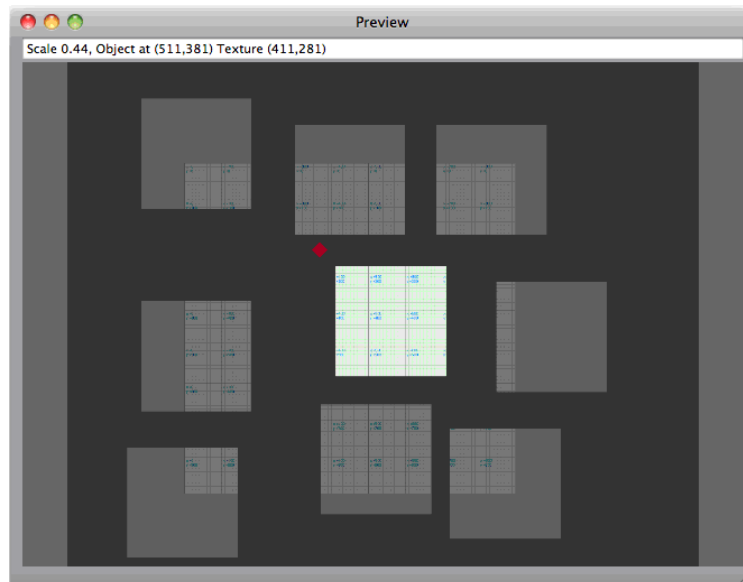
b. Whereas the 2D representation will look like this, with the graph paper moving around, but the outline of the complete MultiScreen object staying the same.
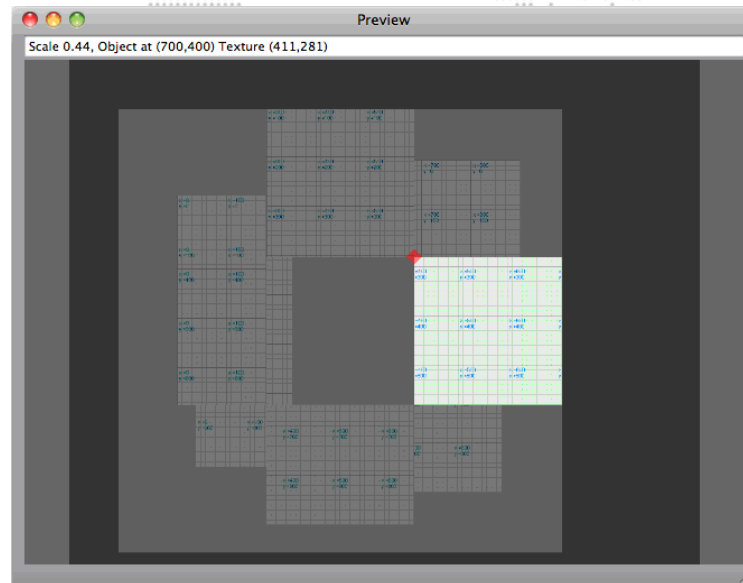
Note: In both the 2D and 3D representations, the pink coloring is used to show a portion of the screen object that has no texture (i.e., where the screen object is fully transparent) due to the either the object's texture origin/width/height or the received control values.

When you first use this revised configuration in Mbox, you'll see something like this:
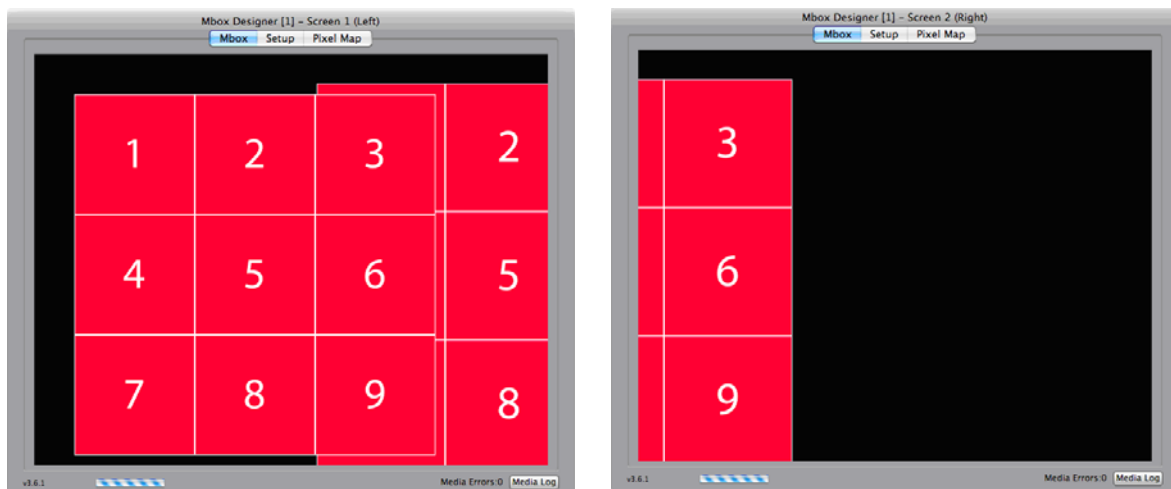
The 3D representation will be positioned across the boundary between the two screens and it will occupy some of the same screen area as the 2D representation. As mentioned previously, you should place the 3D representation onto the second output so it doesn't get in the way of the images being send to the LED panels.

Step 22. To accomplish the move, open the Mbox Remote application, show the Server Preferences window, select the preference called "3D Center Left X," and adjust it so the 3D representation moves and becomes centered on the right-hand screen. A value of "0.6" should be appropriate. (Refer to the Mbox Remote User manual for more information.)

If you adjust the camera Dolly, Orbit X, and Orbit Y controls on the Master fixture, you can manipulate the perspective of the 3D view.
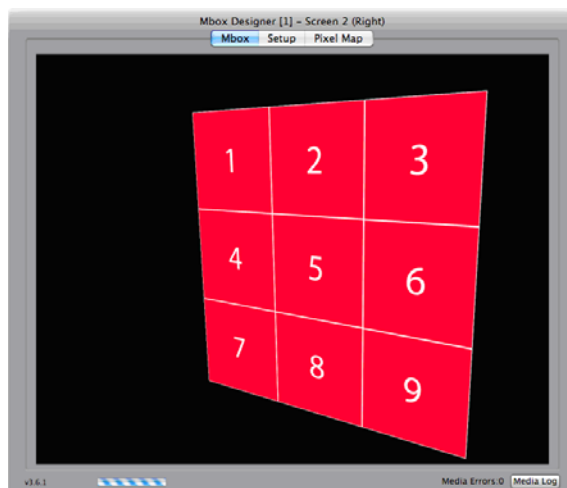
And if you then make some adjustments to the various controls for each of the nine squares, you might end up with something like this:

Note: In the 2D representation, the screen objects don't move but the texture does, resulting in a jumbled image. In the 3D representation, the screen objects move, but the overall texture is not jumbled and is viewable where the screen objects are. The edges of the screen objects are not visible/shown, only the portion of the screen object that has texture on it.

This completes the example tutorial. Now you can go ahead and try something more complicated, or just change some of the values and/or control types in this existing configuration. Adjusting the min and max values for the various X & Y controls may help the result to be more realistic or correct. For a simple control type change, swap the opacity controls to be rotation controls.

> **TIP** You'll probably want to change to an 8-bit 127/128 data type and adjust the scale value if you try changing from opacity to rotation.

Discrete Mapping

It is very easy to take the existing Projected Mapping configuration (created in the "Pixel Remapping Tutorial" on page 87) and make an additional configuration that uses the same screen objects but with Discrete Mapping instead. You can use the copy and paste functionality to make a copy of the configuration created in the tutorial. All you have to do then is remove some controls from each of the screen objects.

Copy and paste the first configuration, then rename it so you can tell it apart from the first. Then, click on the arrows to the left of each screen object to expand and show the controls. With discrete mapping, the texture is going to stay pinned to the screen object no matter where it moves. So all you have to do is delete all controls that deal with texture positioning. (But don't delete the controls that handle the positioning of the screen objects themselves.) If you labeled your controls as in the previous tutorial it will be easy to make the deletions. Select any control that starts with "tX" or "tY" and press the [delete] key. *Be careful though, as there is no undo!*

You might think that you would want to rearrange the Art-Net channels for this second configuration, but you probably don't. The nice thing is that the same external control still controls the same features in both configurations. Since you can switch from one configuration to the other on-the-fly (using the Animation A parameter on the layer) it's easy to use additional configurations at any time. Keeping the same patch for the controls means you don't have to change those values when switching configurations.

ADVANCED CONTROL

The following section covers advanced Pixel Remapping features. If you are unsure about how to use these features, contact PRG support for assistance.

Triangle Strips and Triangle Fans (3D Vertex Array Screen Objects)

In order to support non-rectangular shapes and/or to allow screen objects to be rotated in 3D space, standard screen objects may be converted to Triangle Strip screen objects or Triangle Fan screen objects. Standard screen objects are referenced by their origin and their width and height. Triangle strips and fans are referenced by their vertices.

To convert a screen object to a triangle strip or fan, select the screen object in the listbox on the main window. Then click on the blue pop-up handle at the top-left of the window and select "Convert to Strip" or "Convert to Fan" from the list.

CAUTION! Once you convert a screen object to a triangle strip or fan, you cannot undo the change.

Triangle strips are the simpler of the two triangle-based screen objects. The rectangular screen object is converted to a screen object made from two triangles, as in the second illustration below. Four vertices are created, one for each corner of the screen object. Texture coordinates for the vertices will also be created.

Note: The vertices are not ordered in a circular fashion around the perimeter the rectangle, but rather such that each set of three consecutive vertices describes a closed triangle.

Standard Screen Object   Triangle Strip Screen Object
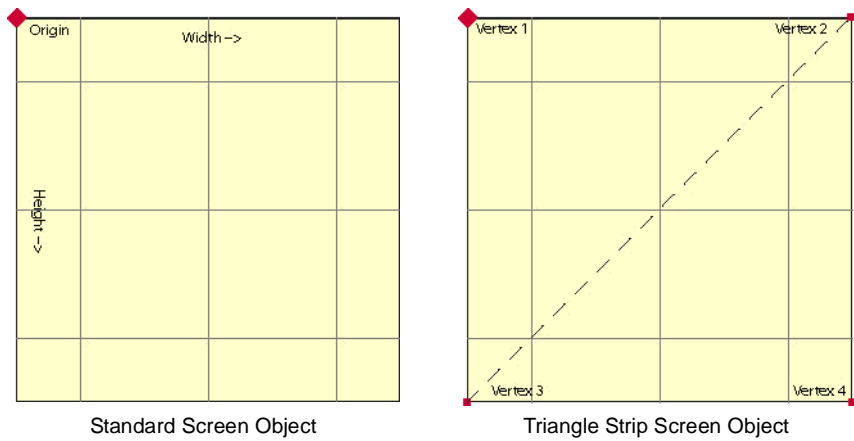
The two illustrations below show the coordinate data for the standard screen object and the triangle strip screen object:

By editing the X and Y coordinates of the triangle strip, you can create differently sized triangles. Editing the tX and tY coordinates adjusts the texture mapping to the vertices of the screen object. Textures will be stretched as required to match the coordinates, be careful, otherwise your texture may be distorted! The + and - buttons allow you to add or delete individual vertices to as to create shapes that are not four-sided.

Triangle Fan screen objects are similar to triangle strip objects, but the ordering of the vertices is different and the origin is used as one point in every triangle. The same screen object from the illustrations above becomes far more complex when converted into a triangle fan. Instead of being two triangles, the object is now four triangles. Each triangle is made up of the first vertex and two other adjacent vertices.

In this manner you can make an object with radial symmetry and with enough vertices to smooth the curve, create an arc or circle.

Screen (3 controls)

☐ Orthographic

X	Y	Z	tX	tY
250	250	0	150	150
100	100	0	0	0
400	100	0	300	0
400	400	0	300	300
100	400	0	0	300
100	100	0	0	0
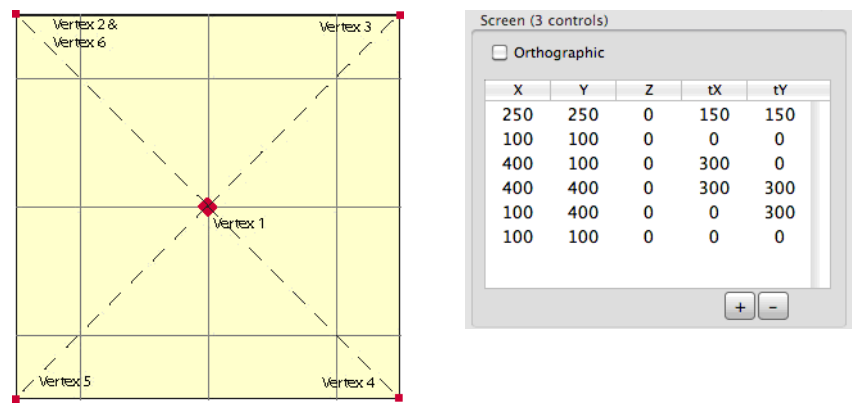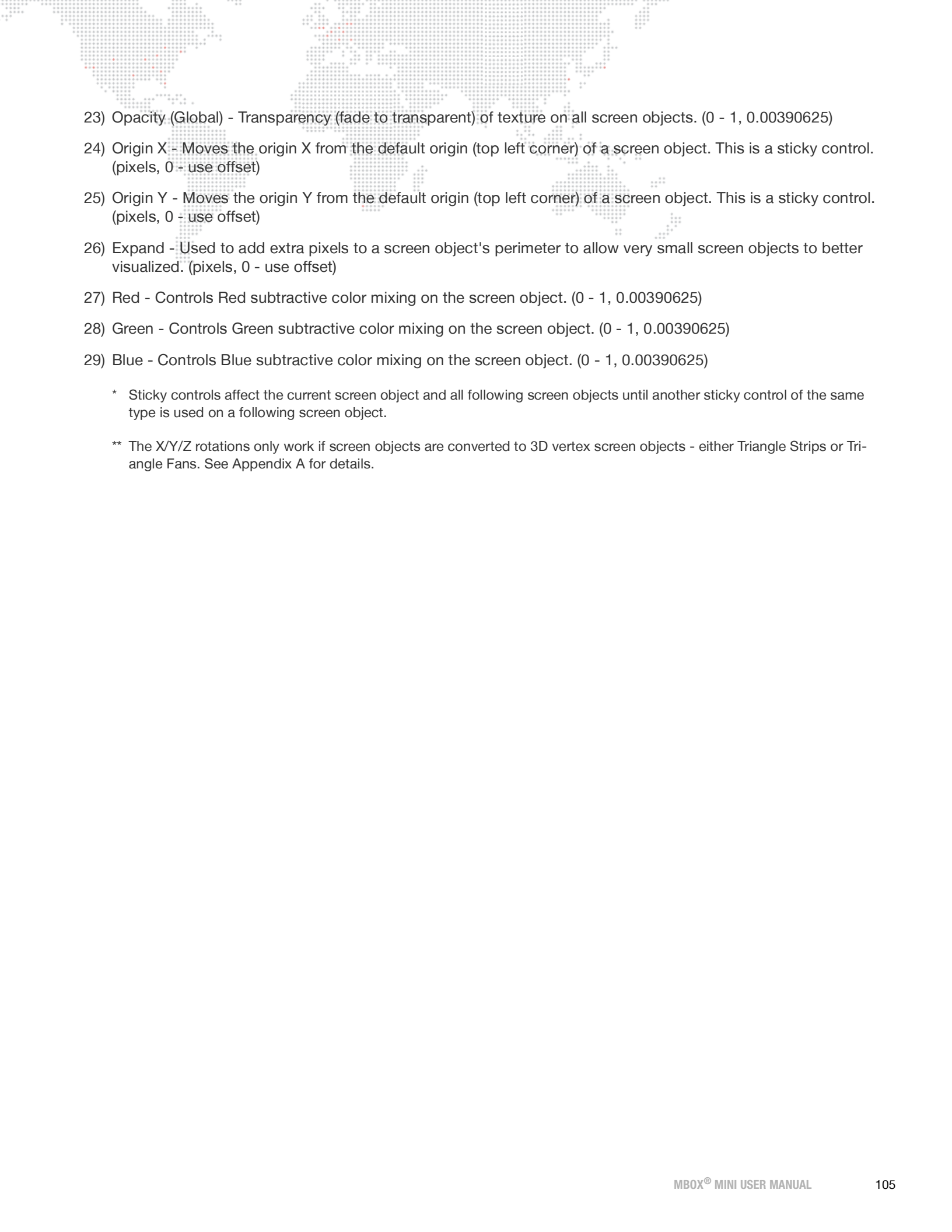
GLOSSARY

Screen Object Control Types (Units, Typical Scale Value)

1) Texture X Position - Left/Right position of the texture on screen object. (pixels, varies)

2) Texture Y Position - Up/Down position of the texture on screen object. (pixels, varies)

3) X Position - Left/Right position of the screen object. (pixels, varies)

4) Y Position - Up/Down position of the screen object. (pixels, varies)

5) Dimmer - Brightness (fade to black, not transparent) of texture on screen object. (0 - 1, 0.00390625)

6) Opacity - Transparency (fade to transparent) of texture on screen object. (0 - 1, 0.00390625)

7) Tex Rot 90 - Texture rotation in 90° increments. This is a sticky* control. (90° steps, 1)

8) Enable - Enable/Disable rendering of screen object - make it visible/invisible. (Enable/Disable, 1)

9) Abort - Enable/Disable rendering of all screen objects after current object. This is a sticky control. (Enable/ Disable, 1)

10) Texture Gain - Multiplier for texture position on screen object. Allows texture position controls to be scaled on-demand for either greater accuracy or increased range. (%, varies)

11) Position Gain - Multiplier for position of screen object. Allows position controls to be scaled on-demand for either greater accuracy or increased range. (%, varies)

12) Tex Rotate - Texture rotation in an arbitrary fashion. (0 - 1, 0.00390625)

13) X Rotate - Rotates Triangle Strip screens or Triangle Fan screens about their X axis, with the screen's origin being the origin of rotation. (0 - 1, 0.00390625)**

14) Y Rotate - Rotates Triangle Strip screens or Triangle Fan screens about their Y axis, with the screen's origin being the origin of rotation. (0 - 1, 0.00390625)**

15) Z Rotate - Rotates Triangle Strip screens or Triangle Fan screens about their Z axis, with the screen's origin being the origin of rotation. (0 - 1, 0.00390625)**

16) Repeat Count - Sets number of automatic repeats of current screen object, typically used with a fixed offset value only, not live control from Art-Net. (# of repeats, 0 - use offset)

17) Repeat X Dist - Sets X position offset distance of each repeat of a repeated screen object, typically used with a fixed offset value only, not live control from Art-Net. (pixels, 0 - use offset)

18) Repeat Y Dist - Sets Y position offset distance of each repeat of a repeated screen object, typically used with a fixed offset value only, not live control from Art-Net. (pixels, 0 - use offset)

19) Repeat tX Dist - Sets texture X offset distance for texture on each repeat of a repeated screen object, typically used with a fixed offset value only, not live control from Art-Net. (pixels, 0 - use offset)

20) Repeat tY Dist - Sets texture Y offset distance for texture on each repeat of a repeated screen object, typically used with a fixed offset value only, not live control from Art-Net. (pixels, 0 - use offset)

21) Repeat X/tX - Sets both X position and texture X position distances for each repeat of a repeated screen object, typically used with a fixed offset value only, not live control from Art-Net. (pixels, 0 - use offset)

22) Repeat Y/tY - Sets both Y position and texture Y position distances for each repeat of a repeated screen object, typically used with a fixed offset value only, not live control from Art-Net. (pixels, 0 - use offset)

23) Opacity (Global) - Transparency (fade to transparent) of texture on all screen objects. (0 - 1, 0.00390625)

24) Origin X - Moves the origin X from the default origin (top left corner) of a screen object. This is a sticky control. (pixels, 0 - use offset)

25) Origin Y - Moves the origin Y from the default origin (top left corner) of a screen object. This is a sticky control. (pixels, 0 - use offset)

26) Expand - Used to add extra pixels to a screen object's perimeter to allow very small screen objects to better visualized. (pixels, 0 - use offset)

27) Red - Controls Red subtractive color mixing on the screen object. (0 - 1, 0.00390625)

28) Green - Controls Green subtractive color mixing on the screen object. (0 - 1, 0.00390625)

29) Blue - Controls Blue subtractive color mixing on the screen object. (0 - 1, 0.00390625)

* Sticky controls affect the current screen object and all following screen objects until another sticky control of the same type is used on a following screen object.

** The X/Y/Z rotations only work if screen objects are converted to 3D vertex screen objects - either Triangle Strips or Triangle Fans. See Appendix A for details.

Notes

6.

VIDEO INPUT

This chapter provides instructions for the Mbox video inputs.

+   EXTERNAL VIDEO SIGNAL INPUT

+   SYPHON VIDEO INPUT

EXTERNAL VIDEO SIGNAL INPUT

Overview

Mbox supports three types of external video inputs: PCIe cards, Blackmagic UltraStudio Thunderbolt, and USB/Firewire devices. For best performance, Mbox has native support for the Blackmagic (Decklink and UltraStudio) and the Active Silicon Phoenix products. Other PCIe cards that support QuickTime streaming may work, but not as well. USB/FireWire devices must output a QuickTime stream.

Mbox Mini allows up to two video inputs per server. The number of inputs will depend on the video input device(s) used. Each of the two inputs must be set up in advance to work properly with Mbox; setting the input type, input number, and video format.

CAUTION! To configure Mbox to work with any capture device, make sure the latest version of the device drivers have been installed.

Blackmagic Decklink and Active Silicon Phoenix Video Capture Cards (PCIe)

Mbox has been optimized to provide the best possible video input (quality, framerate, latency) with Blackmagic Decklink and Active Silicon Phoenix PCIe cards.

Note: At present time Mbox does not support the Decklink 4k capture card, or the Active Silicon LFG cards.

To setup Decklink Extreme in Mac Preferences:

The Decklink Extreme HD card must first be set to receive the correct format (SDI, component, or composite) using the Blackmagic preference pane in the Mac System Preferences. (The Decklink Quad and Active Silicon cards do not require this step.)

Step   1.   Quit Mbox application, if running.

Step   2.   At **Apple** menu, select **System Preferences**.

Step   3.   At System Preferences window, click on **Blackmagic Design** button.
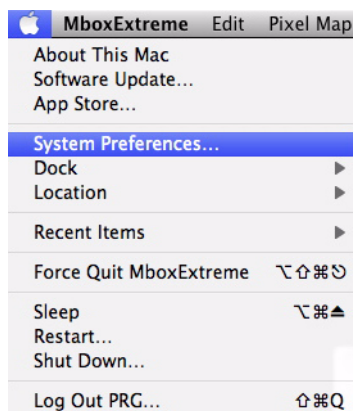
Note: This is an example System Preferences window. The layout will vary for Extreme and Quad setups.
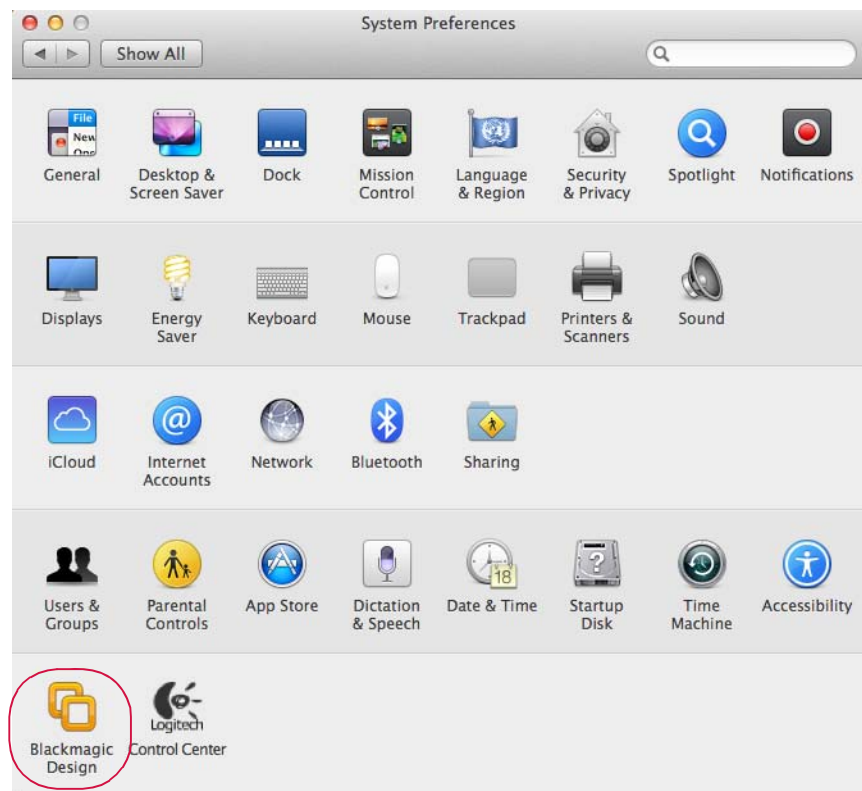
Step   4. At Blackmagic Design Decklink window, click on "Set input" pop-up menu to select an appropriate input type. The three possible selections are: SDI Video, Y, R-Y, B-Y Video, and NTSC/PAL (Y in). These correspond to SDI, Component, and Composite video input types. Because Mbox does not use the audio input on the Decklink Video Capture card, the audio type in each of the inputs can be ignored.

Step   5. Close Blackmagic Design Decklink window.

To configure Mbox video inputs for Decklink or Phoenix cards:

Step   1. Launch Mbox Mini application and go to Mbox Setup tab.

Step   2. At Video Inputs section, click setup button for the index number (Folder 255, File 254 - 255) that you want to assign the video input to.
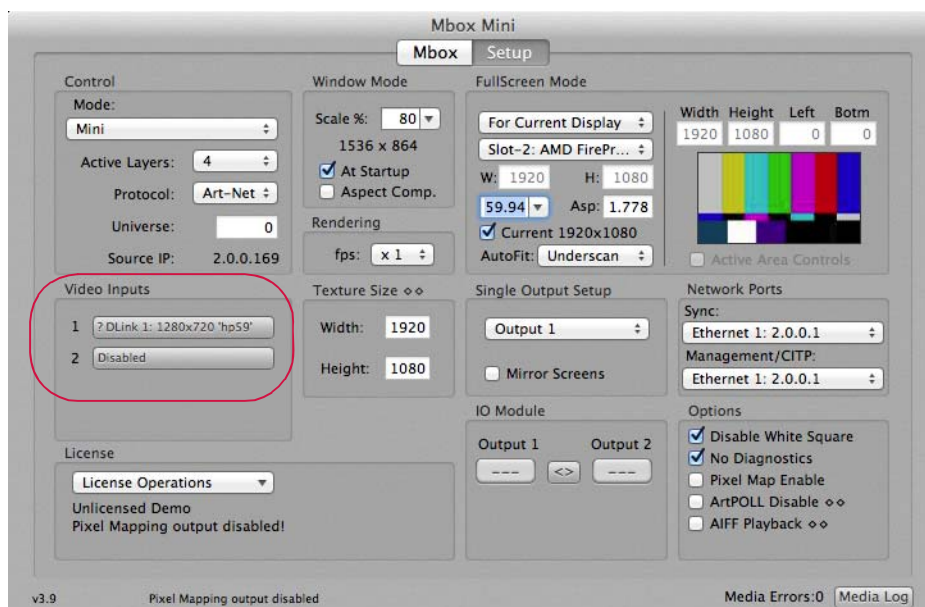
Step 3. Configure Video Inputs window depending on the type of card:

a. **Decklink card**: At Video Inputs window ,set input type pop-up to "Blackmagic," then set input number pop-up to match the input you will be using on the card. Always use 1 for HD Extreme and use 1-4 for Quad. Finally, set the input format pop-up to the correct type.
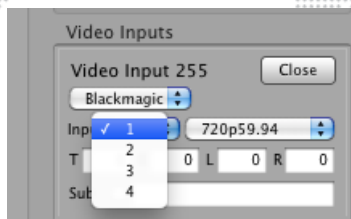
b. **Phoenix card**: At Video Inputs window ,set input type popup to "Phoenix," then set the input number pop-up to match the input you will be using on the card. The first Phoenix card will use inputs 1A and 1B, and a second card will use inputs 2A and 2B. Finally, set the input format pop-up to the correct type.

Step 4. The edit fields labeled T, B, R, and L can be used to enter a crop (in pixels) to the incoming video signal.

Step 5. When enabled, the Sub checkbox will cause Mbox to display a substitute image if a video input is not available. The default substitute image is a blue rectangle with the input type number and format listed. You can use a custom substitute still image by typing the Folder.File index (e.g. 004.001) or the full path for the file (e.g. /Users/username/Documents/Raster.jpg) into the field next to the Sub checkbox.

Step 6. Once everything is set up correctly, click **Close** button to close setup window.

Step 7. Make sure Texture Size width and height settings are as large or larger than the camera input resolution width and height. Failing to set the Texture Size correctly will result in an unusable image when the video input is requested for display. Texture Size is entered at the Setup tab of the Mbox Setup window.

Step 8. In some cases, you may need to restart the Mbox software after changing the video input settings.

Blackmagic UltraStudio Devices (Thunderbolt)

Mbox also supports the UltraStudio Mini Recorder, Express, and 4K. To obtain the best performance with these products, make sure that the Thunderbolt bus has enough available bandwidth for the capture device and any other connected Thunderbolt devices. Latency with Thunderbolt devices may be slightly higher than that obtained with PCIe cards.

Note: At present time Mbox does not support input resolutions larger than 1920 x 1080 from UltraStudio devices.

To setup UltraStudio:

The UltraStudio must first be set to receive the correct format (SDI, composite, component or HDMI) using the controls on the device (for the 4K) or the Blackmagic preference pane in the Mac System Preferences (for the Mini Recorder and Express). Use the following steps if it is necessary to configure in Mac System Preferences:

Step   1.   Quit Mbox application, if running.

Step   2.   At **Apple** menu, select **System Preferences**.

Step   3.   At System Preferences window, click on **Blackmagic Design** button.
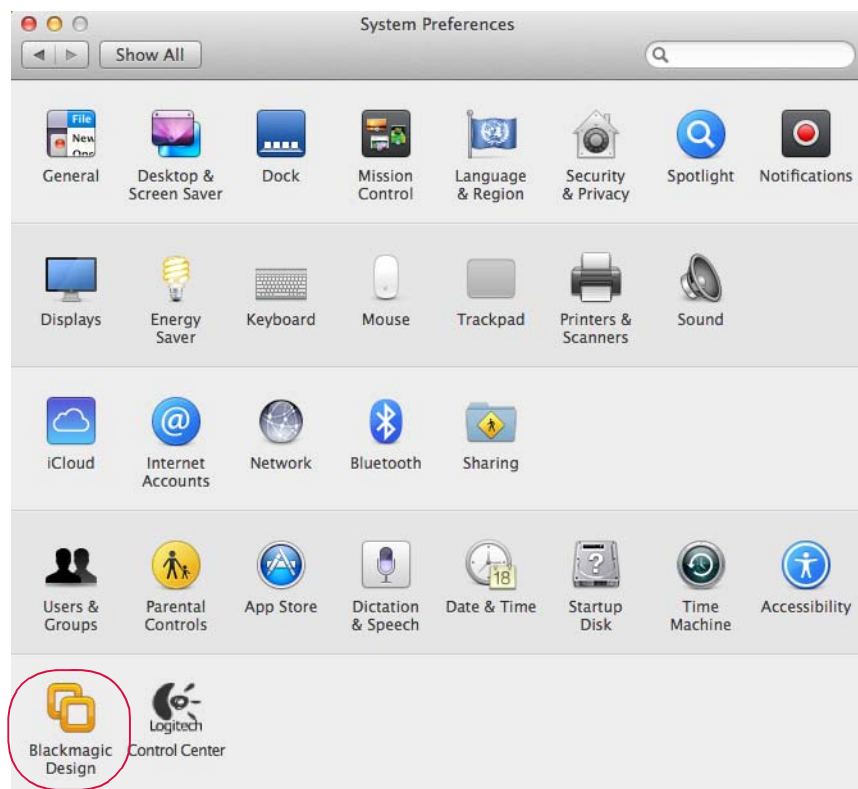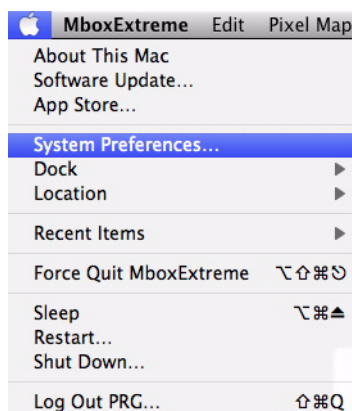
Step   4. At Blackmagic Design Decklink window, click on "Set input" pop-up menu to select an appropriate input type. The possible selections may include SDI Video, HDMI Video, Y, R-Y, B-Y Video, and NTSC/PAL (Y in), and will depend on the UltraStudio device installed. Because Mbox does not use the audio input on the UltraStudio device, the audio type in each of the inputs can be ignored.
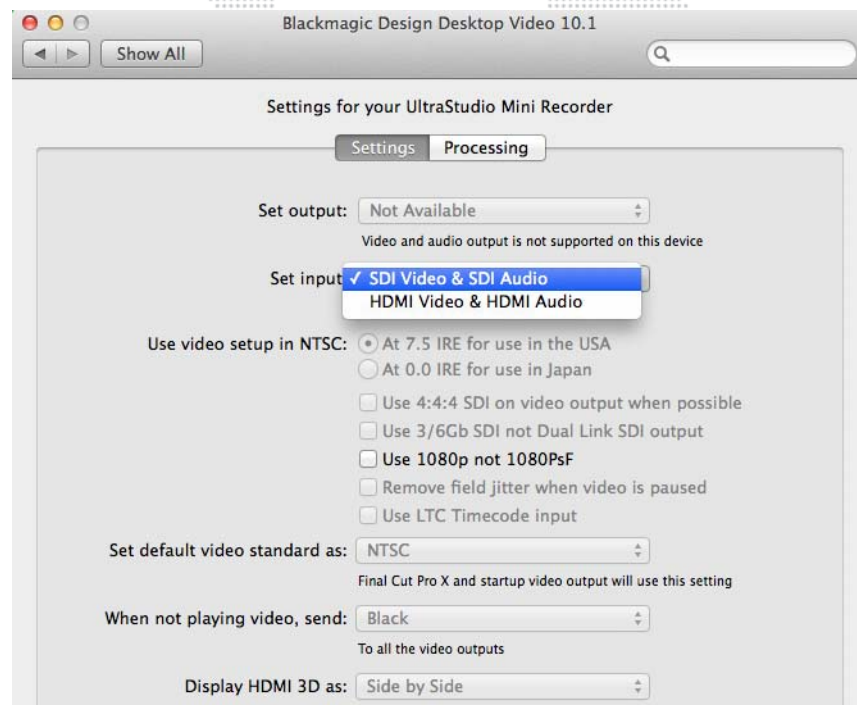
Step   5. Close Blackmagic Design Decklink window.

To configure Mbox video inputs for UltraStudio card:

Step   1. Launch Mbox Mini application and go to Mbox Setup tab.

Step   2. At Video Inputs section, click setup button for the index number (Folder 255, File 254 - 255) that you want to assign the video input to.
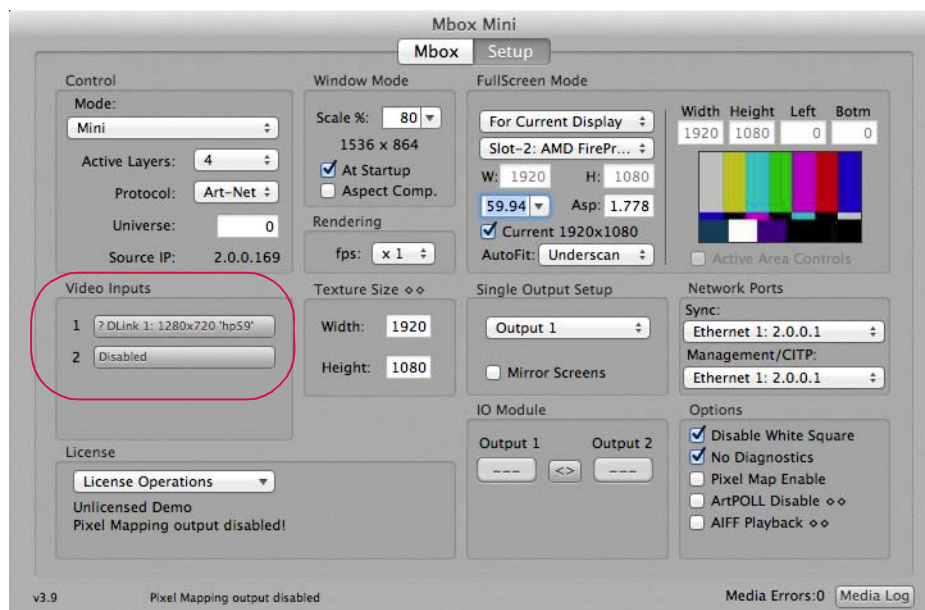
Step 3. At Video Inputs window, set input type pop-up to "Blackmagic," then set input number pop-up to input 1. Finally, set the input format pop-up to the correct type.

Step 4. The edit fields labeled T, B, R, and L can be used to enter a crop (in pixels) to the incoming video signal.

Step 5. When enabled, the Sub checkbox will cause Mbox to display a substitute image if a video input is not available. The default substitute image is a blue rectangle with the input type number and format listed. You can use a custom substitute still image by typing the Folder.File index (e.g. 004.001) or the full path for the file (e.g. /Users/username/Documents/Raster.jpg) into the field next to the Sub checkbox.

Step 6. Once everything is set up correctly, click **Close** button to close setup window.

Step 7. Make sure Texture Size width and height settings are as large or larger than the camera input resolution width and height. Failing to set the Texture Size correctly will result in an unusable image when the video input is requested for display. Texture Size is entered at the Setup tab of the Mbox Setup window.

Step 8. In some cases, you may need to restart the Mbox software after changing the video input settings.

QuickTime Streaming Devices (PCIe or USB/FireWire)

Mbox also supports all other capture devices that conform to the QuickTime streaming format. Such devices include USB and FireWire cameras, and some PCIe capture cards.

Setup on Mac:

QuickTime streaming devices may require driver software to be installed. Check with the manufacturer and install the most current version. Additional setup may be required depending on the device.

To configure Mbox video inputs for QuickTime streaming device:

Step   1. Launch Mbox Mini application and go to Mbox Setup tab.

Step   2. At Video Inputs section, click setup button for the index number (Folder 255, File 254 - 255) that you want to assign the video input to.
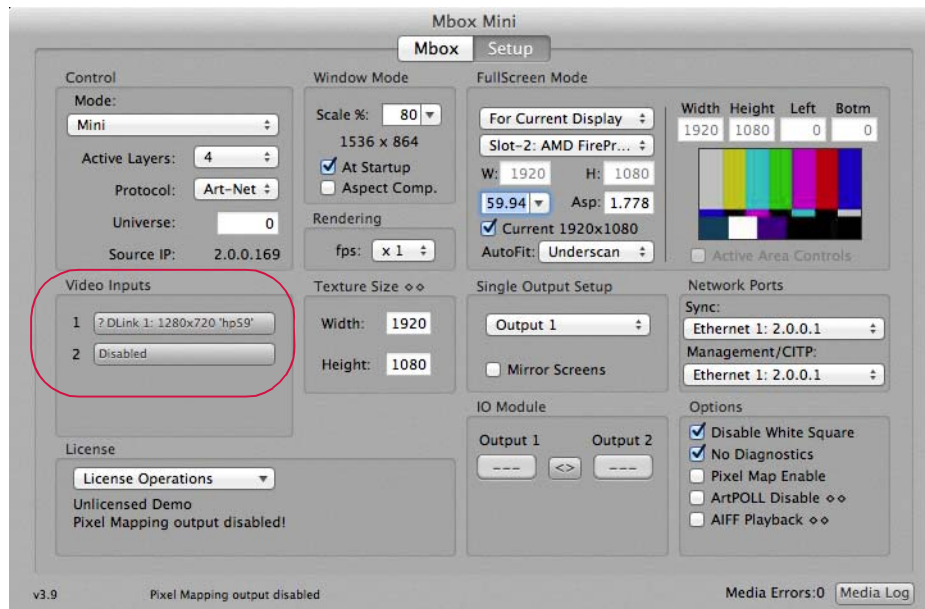
Step   3. At Video Inputs window, set input type pop-up to "QuickTime," then select the device by name on the input pop-up.

Step   4. The edit fields labeled T, B, R, and L can be used to enter a crop (in pixels) to the incoming video signal.

Step   5. When enabled, the Sub checkbox will cause Mbox to display a substitute image if a video input is not available. The default substitute image is a blue rectangle with the input type number and format listed. You can use a custom substitute still image by typing the Folder.File index (e.g. 004.001) or the full path for the file (e.g. /Users/username/Documents/Raster.jpg) into the field next to the Sub checkbox.

Step   6. Once everything is set up correctly, click **Close** button to close setup window.

Step 7. Make sure Texture Size width and height settings are as large or larger than the camera input resolution width and height. Failing to set the Texture Size correctly will result in an unusable image when the video input is requested for display. Texture Size is entered at the Setup tab of the Mbox Setup window.

Step 8. In some cases, you may need to restart the Mbox software after changing the video input settings.

SYPHON VIDEO INPUT

Using Syphon

Mbox Mini includes the Syphon framework, which allows Mbox to either send or receive real-time, full-framerate videos or stills between applications on the same machine. For example, it is possible to place the video output from Quartz Composer, Unity 3, or other applications onto a layer within Mbox.

Note: For more information on the Syphon framework, refer to the webpage: http://syphon.v002.info

---

Without any configuration, Mbox can display Syphon inputs in the order they became active on the computer. This allows for quick use without setup, but if more than one input is present, it does not guarantee that the same input source will always have the same index number for selection purposes.

You can use Mbox Remote's remote preference setting feature to assign a specific Syphon input to a specific index. To do this, enter the configuration options into the "Syphon Sources" text field in the following syntax:
<Application Name>:<Source Name>
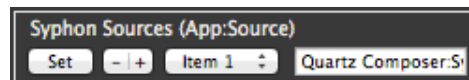
In the above example, the full text is "Quartz Composer:Syphon Output." In this case Quartz Composer is the application presenting the source to Mbox and the individual source's name in Quartz Composer is "Syphon Output." This Syphon input has been assigned to Item 1, which relates to Folder 255 and File 201. (See Syphon Input section below.)

Finding Application and Source Name

The Mbox logging system can be used to find the correct application and source name for any active Syphon source.

Step   1. To access this information, first start the Syphon source application, then start Mbox.

Step   2. With Mbox selected in Window mode, press [⌘ L] to display the current Mbox Log.

Step   3. In the console Logging application, enter "Syphon:" into the search filter field at the top-right to display only Syphon information. The first available Syphon slot will display the App:Source pair, if it is being presented on the system.
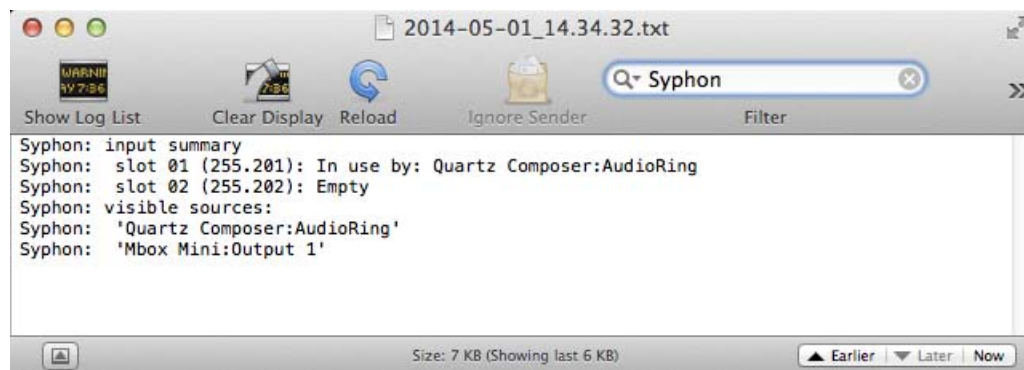
Step 4. Once you have identified the application and source names, copy and paste the App:Source pair into a Syphon Sources item in Remote Preferences to reserve that source for a specific Syphon input (refer to the Mbox Remote User manual for more information). When Mbox next finds that App:Source pair, it will automatically allocate it to the configured Syphon input.
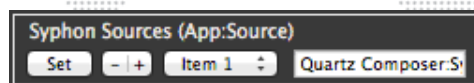
Guidelines:

+ Mbox Mini has a limit of 2 Syphon inputs.

+ Mbox will assign a non-configured Syphon input to the next available slot on startup. Keep in mind, if no slot reservation is given for an input then the source will not necessarily be attached to the same slot if more then one source is available.

+ Syphon inputs are received at the same size they are rendering in the source application. Therefore, Mbox's Texture Size setting must be at least as big as the source image in order for it to be displayed correctly.

Syphon Input

The table below relates the Syphon Slot and Item numbers to the media folder and file that will be assigned to the syphon input:

Folder.File	Function
255.201	Syphon Item 1
255.202	Syphon Item 2

Syphon Output

Mbox can also use Syphon to output video to other Syphon-aware applications on the same computer (e.g. MadMapper). In this way, Mbox's output can be used as an input to the other application. This might be used for further manipulation, 3D mapping, etc.

Syphon output works automatically. Mbox creates and advertises the Syphon sources. You can then launch the other application, look for Syphon sources, then select the one you want to use.

Mbox Mini has up to 2 outputs; one for each possible output.

7.

CONTENT CREATION

This chapter contains information on the proper formatting of content for playback with Mbox.

+   CREATING CUSTOM CONTENT

CREATING CUSTOM CONTENT

Creating Movies and Audio Files

As mentioned earlier in this manual, Mbox will try to play any QuickTime movie -no matter what codec it may be - however, movies using non-preferred codecs may not play as well as movies that use the preferred codecs. While Mbox will do the best job possible to play any movie content, if the file is not optimized specifically for Mbox, poor results may occur. There are many elements of movie files that need to be considered when creating content: file extension, codec, compression, frame rate, resolution, and audio. The following sections take an in depth look at each of these requirements.

File Extension

Mbox can open any file that QuickTime can open. However, Mbox will only open movie files with either a .mov or .vid file extension unless the application is told to use other extensions, such as mp4, .qtz. By default, some of these extensions have been enabled, but at times, additional extensions may also need to be enabled. Keep in mind that any movie file that does not have a .mov or .vid file extension is treated as a special case for playback, and as such, may respond differently to playmodes, effects, and transitions. It is a good idea to always test such files in advance. In order to add file extension types to Mbox, use the Mbox Remote application that is included in the Mbox Utilities folder. (Refer to the Mbox Remote User manual for more information.)

Codec

Beyond the file extension, the movies' codec (**co**mpressor/**dec**ompressor) is very important. Mbox will try to play any QuickTime movie, no matter what its codec is. However only movies using the Photo-JPEG, Apple Intermediate, Apple Animation, Apple ProRes, or DV codecs will receive preferred treatment and be able to take advantage of all of the Mbox functionality. Photo-JPEG should be the first choice when creating content for Mbox. Apple ProRes and DV will work very well, although there are some additional considerations that must be taken into account when using these codecs. Any other codec will be treated as "non-preferred" and may not play as well as a movie that uses a preferred codec.

+ **Photo-JPEG** is a high quality, scalable format that does not use any temporal compression. As such, it is perfect for use with Mbox. These movies can be of any resolution (within performance limits) and will take full advantage of the many Mbox playmodes. Photo-JPEG is a processor-intensive codec, so care must be taken to keep movies to a reasonable quality setting otherwise playback will not be smooth. Typically, a setting of Medium or 50%-60% is appropriate for Mbox.

+ **Apple Intermediate** is a scalable codec that uses spatial compression. In many ways it is similar to Photo-JPEG, but tends to place a lighter load on the processor.

+ **Animation** is one of the two codecs that allow for an alpha channel which can be used by Mbox - the other being Apple ProRes 4444. The only reason to use the Animation codec is when a movie requires transparency. In this case, be aware that the bit rate of Animation movies can be very high as its compression scheme is actually optimized for large areas of the same color - essentially for animated cartoon content. To play movie files with alpha channel information, the Mbox Movie Converter application provides a method of re-rendering Animation codec movies to a format that may be easier to play.

+ **Apple ProRes** is a high-quality codec that was designed to compress HD content sufficiently to keep bit rates at SD levels. Like Animation, the bit rate of ProRes content can be very high, so care should be used when creating movies in this format.

+ **DV** is a compression format typically used by video cameras. For best results, DV-encoded movies must conform to either NTSC or PAL resolutions, and therefore, should not use any random resolution. The quality of DV is typically not as high as that of Photo-JPEG, although it is less processor-intensive. DV movies from a camera or DVD source will often be interlaced. If you have an interlaced DV movie file, you can change the file extension from .mov to .vid and Mbox will deinterlace the file as it plays. However, better quality may result by converting/re-rendering such movies to Photo-JPEG in advance. DV movies may also have embedded audio tracks that should be separated before attempting to play the movie on Mbox.

+ **Other codecs**, when treated as non-preferred, are played in a QuickTime "wrapper" rather than natively in Mbox. Because of this, playback may be poor, playmodes other than FWD Loop and FWD Once may not work, movies may not loop unless the in/out points are rolled inwards slightly, frame blending will not work, and some transitions and effects may not work properly. On the plus side, embedded audio tracks in movies with non-preferred codecs may play when the movie is played. To force Mbox to attempt to play non-preferred codecs natively, however, this is not recommended. Please contact your PRG representative if you would like further information about this feature.

Compression (Bit Rate)

Movie compression can have a large impact on playback performance. There is a balance between quality and performance that can be tipped in either direction by adjusting the amount of compression when a movie is rendered. Typically, compression is labeled as "Compressor Quality" or something similar and the scale goes from "least" to "best" or 0% to 100% - more compression to less compression. The more compression used, the smaller the movie file will be and the bit rate (or data rate) of the movie will be lower. Lower bit rates theoretically lead to better playback performance, but below a certain point the quality may be unacceptable. Alternately, lowering compression will provide a higher quality movie, but due to a higher bit rate, playback performance may suffer. At a certain limit in either direction there will be a point of diminishing returns where either quality or playback is unacceptable. Typically, Mbox will provide the best performance to quality ratio at Medium or 50%-60%. Higher quality settings may not look noticeably better for the trade-off in performance. As a rule of thumb, movie bit rates should be kept below 65Mbits/sec in order to play content on multiple layers at the same time. In special circumstances, Mbox can handle bit rates up to approximately 200Mbits/sec.

Frame Rate

Movie frame rate may also need to be considered when creating content for Mbox. For optimal results, content should be rendered at a frame rate that is a divisor of the current Fullscreen mode output frequency. Therefore, if the Fullscreen mode has a 60Hz refresh, movies that play at 30fps will synchronize nicely with the output of the Mac Graphics card (each frame of the movie will be repeated twice to equal two fields of the output). While content with lesser frame rates will work, there may be some noticeable effects. In the above example, if the movie has fewer than 30fps, then the Graphics card will need to repeat some frames more than twice and the movie may appear to play less smoothly. Depending on the frame rate of the movie, this effect may be more or less noticeable. If the movie's frame rate divides evenly into 60 then the movie will play more smoothly; so 20fps content will look better than 24fps content. So if the Mbox output frequency is 50Hz, then it would be best to create content that is rendered at 25fps. If output frequency is 59.94Hz, then content should be rendered at 29.97fps, and so on. (Refer to "Setup: Fullscreen Mode" on page 38 for more information about setting the output frequency.)

Resolution

As noted above, movie resolution has an impact on the performance of the Mbox server. To put it simply, the larger the movie's pixel dimensions, the harder it is to play. For best results, always try to play the smallest movie possible. Furthermore, always make sure that the Mbox texture size setting is no larger than the largest piece of content that you intend to play. Setting the texture size larger than necessary will affect movie playback. Of course there are factors outside of Mbox that need to be taken into consideration when choosing a content resolution - type of output signal, display device native resolution, size of projection surface- but for the most part, there is not much point in playing content that has a higher resolution than the selected Fullscreen output size. But admittedly, sometimes the smallest possible movie is not the one that looks the best with your particular setup, so make sure to test in advance if possible.

Audio

Audio file manipulation for Mbox is one of the more complicated tasks in content creation. Particularly, AIFF audio files that are associated with movie files rather than audio that is embedded in movies that use non-preferred codecs. Remember that audio playback cannot be sped up or slowed down when using associated AIFF files, and that audio files will loop when the movie hits its out-point. So it is imperative to make sure that an audio file's playback rate matches the movie's default playback rate, and that the audio file is exactly the same length as the movie. The best way to do this is to keep the audio and video components of movies together until the very last minute (all throughout the editing process). Once the editing process is complete, then separate the two files. Try to avoid any manipulations

of either file (especially the movie file) once the two have been separated. (You can always shorten the movie or play portions of it by adjusting the in and out points.)

AIFF audio files for playback on Mbox must be encoded as 2-channel (Stereo: L & R), LPCM, 16-bit Integer (Big or Little Endian) at a sample rate of 44.1 kHz. Other sample rates will play, but will not play at the correct speed. Mbox cannot play more than one AIFF audio file at a time. The most recent file requested will play. There is a slight crossfade between tracks that play back, but this is only there to eliminate any pops or clicks when the audio switches files. On the other hand, if you choose to play the audio that is embedded in movies (refer to "Audio" on page 72 for further info) you can play more than one audio track at a time. However, this can get "messy" since there is no crossfade or volume control when playing embedded audio.

The audio volume control channel on the Master fixture can be used to modify the volume of AIFF audio file playback. However, at present time the volume control does not affect embedded audio tracks in movies that use a non-preferred codec.

Discrete Audio Support for Multichannel Audio

FireWire devices require audio outputs to be assigned to channels called "discrete - 0" through "discrete - 31." Most QuickTime movies are made with mono, stereo, or 5.1/7.1 audio and the audio tracks and/or channels must be manually re-assigned to work properly with the FireWire interfaces. To simply this, Mbox has the ability to automatically take audio tracks in movies and re-assign their output to the discrete channels used by FireWire audio interface boxes (e.g., MOTU, Fireface, RME).

With the Discrete Audio preference enabled, Mbox will automatically re-assign audio channels (sequentially) to the discrete range if it encounters a movie with more than one audio track or a track with more than two channels. This means that movies with a single mono or stereo track will not be affected, but since the mono output is always assigned to discrete - 0, and stereo L & R to discrete 0 & 1, this is not an issue.

To enable discrete audio support:

Step   1.   Use Mbox Remote application to enable the preference named "Audio Discrete Mapping." (Refer to the Mbox Remote User manual for more information.)

Step   2.   If a movie is currently playing, choose a different movie on that layer and then return to the original movie.
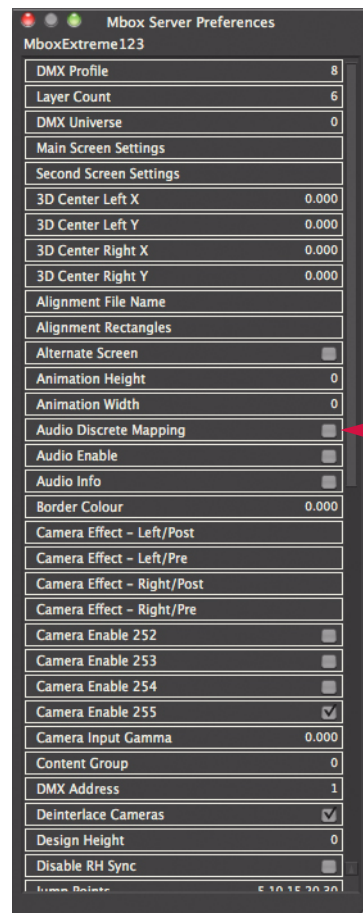
Alpha Channel in Movies – the Animation and ProRes 4444 Codecs

It is often very useful to be able to add areas of transparency to a layer when using Mbox. This will make it possible to see other layers behind, to create a "knockout" for IMAG, or to properly display a movie that uses green-screen technology.

When creating content, there are several QuickTime codecs that allow an alpha channel (transparency information) to be embedded in a movie. For the purposes of Mbox, the only useful codecs are Animation and ProRes 4444. Encoding content using the Animation or ProRes 4444 codecs can result in movies having a very high or unplayable bitrate. Take care to choose the appropriate quality setting to keep the movie's bitrate lower than ~150mbits/sec.

If a movie is created using the Animation or ProRes 4444 codec and played on Mbox, the preferred playback engine will handle it. Mbox is able to use the alpha channel information in the movie and to display transparent areas as transparent. However, unless the movie is quite small and/or has a low bit-rate (< 80mbits/sec), the movie may not play very smoothly, and all other playback on the server may be affected to some extent.

Another way to create a movie with an alpha channel is to use the Animation codec conversion built into the Mbox File Converter application. This application will convert the movie into a format that Mbox is better able to play while still retaining the alpha channel information. When the movie is played in Mbox, the bit-rate is significantly lower – allowing for better playback – and the transparent areas are still transparent.

Applications for Content Creation

There are many applications that can be used to create movie content for Mbox. They vary from the higher end (Adobe After Effects®, Apple Final Cut®) to the lower end (QuickTime® Pro). All of them have their uses, but there are some guidelines and caution to be observed:

+ Never create a movie with compressed headers or prepare a movie for internet streaming.

+ To modify both the framerate of a movie and trim the movie, do not use QuickTime Pro, as this will modify the headers of the QuickTime file to the point where Mbox cannot play it. However, QuickTime Pro can be used to convert a movie from one codec to another or to trim an existing movie. It is recommended that you perform more complex tasks (changing the framerate) in such a way that the movie's source elements are re-rendered into a new file. This is especially critical if audio is involved.

Training Video: Mbox Media Server Training Video Chapter 20: Codes, Storage, and Media Types

Notes

A.

NETWORKING FOR MBOX

This appendix provides guidelines for setting up Mbox networks.

+ MBOX NETWORKING GUIDELINES

MBOX NETWORKING GUIDELINES

Concepts

The majority of communications to and from Mbox use Ethernet protocols. This means Ethernet networks route the data into the Mbox server and from the server to other connected devices. Because all of these protocols need to live in harmony, steps need to be taken to avoid interference between protocols, and to maximize the available network bandwidth/speed. Due to these requirements, it is recommended that the computer used as an Mbox server have more than one network interface. For more complex setups, three or four network interfaces are preferable.

While all the network interfaces on a computer could be connected to the same network to gain some amount of additional bandwidth, doing so would bypass the desire to keep certain protocols separate. Therefore, it is far more useful if each network interface on the computer is configured for and connected to a separate physical Ethernet network - or, if connected to the same physical network, then with a system of VLANs enabled to segregate different protocols.

Connecting more than one network interface to different networks is only the first step. Each interface needs to have its IP Address and subnet mask set correctly for the network that it is being connected to. Getting the IP Address and subnet mask correct ensures that communications to and from the computer will use the correct interface and arrive at the intended destination without any ambiguity or loss of data.

Concept definitions:

+ **Network Interface** - the physical Ethernet connection (typically RJ-45) on the computer that may be used to connect the computer to a physical network.

+ **Network** - a group of devices physically connected to each other that can potentially communicate using Ethernet IP protocols. The word "network" describes not just the physical connection of multiple devices, but also the "web" of electronic communication between them.

+ **Subnet (subnetwork)** - a logical subdivision of a network such that multiple interfaces can be connected to the same physical network, but are placed on different subdivisions of that network for the purposes of routing or for keeping some communications separate. (Note: a subnet is not the same as a VLAN.)

+ **IP Address** - an interface property using the dot-decimal (e.g., 2.0.0.123) notation, where each of the four numbers separated by decimal points is referred to as an octet (because each is an 8-bit number). In combination with the interface's subnet mask, the octets in the IP address designate either the device's sub-network or the device's unique identifier on that subnet.

+ **Subnet Mask** - a second interface property, also using dot-decimal notation (e.g., 255.0.0.0). The value of each octet in the subnet mask determines which of the octets in the device's IP address specify the sub-network and which specify the interface's unique identifier on that sub-network.   The subnet is specified by setting one or more of the octets to a value other than 0.

Subnet Masking Basics

For pretty much all communication to and from Mbox, each octet in an interface's subnet mask will be set to either 0 or 255. Values other than 0 or 255 can be used for more complicated subnetting, but are typically not needed with Mbox and won't be discussed here.

In order to understand how an interface's subnet mask works, you need to look at it together with the interface's IP address. If an octet in the subnet has a value of 255, then the corresponding octet in the interface's IP address is used to designate the subnet that the interface belongs to. If an octet in the subnet mask has a value of 0, the corresponding octet in the IP address is used as part of the interface's unique identifier on that subnet. Within the structure of sub-networks and unique identifiers, it is possible to have multiple interfaces on the same subnet, each being uniquely identifiable. Data can be sent to all interfaces in the subnet by broadcasting, to groups by multicasting, or to individual members by unicasting.

How the IP Address and Subnet Mask Work Together

For a device with the IP address of 192.168.0.123 and a subnet mask of 255.255.0.0, the first two octets of the IP address "192.168" are used to specify the sub-network, and the last two octets "0.123" specify the device's unique identifier within that network.

		Network		Unique ID	
Subnet Mask:	255	255	000	000
IP Address:	192	168	000	123

Depending on the number of octets with a value of 255 in the subnet mask, different descriptions may be given to the interface. A subnet mask of 255.0.0.0 leads to a description of "/8" for the interface. Combined with the interface's IP address, this information would be written as "192.168.0.123/8". A subnet mask of 255.255.0.0 gets a description of "/16" and 255.255.255.0 a description of "/24".

Note: You can't have a /32 subnet mask, as the last octet in an interface's IP address must be used as a unique identifier.

Routing Ethernet Communications

+ **Full Broadcast** - data packets that are sent to the full broadcast destination IP address (255.255.255.255) will be received by all devices on the physical network provided the packets are not blocked by a switch or router between devices. Full broadcast is akin to shouting in a crowded room. Everyone can hear the yelling, but must decide whether to listen or not. Full broadcast is simple and guarantees for the most part that every device on the network can get the data. However, it places a lot of data onto the network, thus using up more of the available bandwidth. Another downside is that receiving devices may need to spend more time than is desirable to filter out data that they don't need to listen to.

+ **Subnet Broadcast** - packets that use a subnet broadcast IP address (x.255.255.255, x.x.255.255, or x.x.x.255) will only be received by interfaces on the matching subnet - i.e., those with a subnet mask that places them into the same subnet. This is one step down from shouting, as subnet filtering allows receiving devices to immediately discard any subnet broadcast data that isn't destined for their subnet.

To obtain the subnet broadcast address for a particular subnet:

Step   1. Take the IP address for a device on that subnet and the subnet mask for that device.

Step   2. Line the address and subnet mask up one above the other, in four columns with each octet being a column.

Step   3. In columns where the subnet mask has a value of 255, write down the value from the IP address' octet.

Step   4. In columns where the subnet mask a value of 0, write down a value of 255.

IP:	192.	168.	0.	123
Mask:	255.	255.	0.	0
Subnet B'cast Address:	192.	168.	255.	255

+ **Unicast** - packets using a specific, unique destination IP address. This is most comparable to two people having a single face-to-face conversation without anyone else able to hear them. Interfaces can have more than one unicast communication at a time. Unicasting cuts down on network overhead as it places the minimum amount of data on the network and because packets are addressed specifically, it eliminates the requirements that a device filter out data it doesn't need to listen to.

+ **Multicast** - packets that are sent to an IP address belonging to a predetermined range of IP addresses. Interface's may join a multicast group to be able to send and receive packets within the group. This is comparable to a group of listeners who all choose to listen to one speaker, or to other members of the group who take on the role as speaker.. The conversation doesn't leave the group.

It is possible for interfaces to send packets to or receive packets sent to a multicast address without joining the corresponding multicast group. But unless the interfaces join the multicast group, switches between the devices may not forward the multicast packets correctly.

Most of the time you can use a /8 subnet mask (255.0.0.0) with Mbox. When you use multiple interfaces on the same computer, with each having a /8 subnet mask, a general rule is to make sure that each interface has a different first octet. Otherwise there may be ambiguity when trying to determine which interface to use for various communications. While all interfaces on a computer may receive packets from any source, packets sent by the computer almost always should leave through only one interface. If that interface is not on the correct subnet (or indeed the correct physical network) then the packets will never get to the intended destination.

For example, you can have two interfaces set up with IP addresses of 2.0.0.1 and 2.1.0.1 where both have a subnet mask of 255.0.0.0. -> Both interfaces are on the same sub-network (i.e. 2.x.x.x), but have unique IDs on that network. Because this is a valid IP scheme, the computer will allow this; but using this setup will cause issues with Mbox software. One or both of these interfaces should use a /16 subnet mask instead of the /8 subnet mask, thereby placing the two ports onto different subnets. -> 2.1.x.x and 2.0.x.x

With all protocols, especially those detailed below, it's best that one and only one port on the computer be on the same subnet as the destination devices. When the computer is told to output data to a specific destination device it avoids confusion if only one port can "talk" to that device.

Standard Control Protocols

Art-Net

Art-Net packets can be fully broadcast (i.e., not restricted to a particular subnet) which makes a device's subnet inconsequential. Some Art-Net devices use full-broadcast to transmit their data, potentially flooding the network with data.

To reduce overhead, most Art-Net devices use a subnet mask of 255.0.0.0 and generally use either 2 or 10 as the first octet in their IP address. This places these devices into a specific subnet and they only talk to other Art-Net devices on the same subnet. Art-Net data would be sent to the subnet broadcast IP address - either 2.255.255.255 or 10.255.255.255. While this cuts down on some bandwidth usage vs. a full broadcast, it should be noted that subnet broadcasts still aren't the most efficient use of bandwidth.

For the best efficiency possible, Art-Net can be unicast from device to device. This cuts down on overhead, but means that the IP address for each device must be known in advance and the precise destination for every piece of data must be specified. This can make the hardware configuration and software setup more complex. Additionally, the device to be unicast to must still be on the same subnet as a port on the Mbox computer, otherwise the packets won't be received at all.

sACN

sACN packets are typically multicast, which sidesteps the restrictions of subnets and subnet masks meaning that any interface can send or receive data for any multicast group it joins. Of course, the interface must still be on the correct physical network for the multicast group. Joining a multicast group is most important when receiving data. Switches can monitor multicast data on the network, and then filter out data that a connected interface has not elected to receive (i.e., for a multicast group it has not joined).

Any interface on the computer can be used for multicast, but as noted earlier, the interface has to be specifically identified and registered as a member of the appropriate multicast group. sACN uses predetermined multicast groups and therefore the address for each group is preset as well. With its registering as a member of all of the necessary groups (for transmitting or receiving), Mbox will not be able to send or receive sACN packets. For the purposes of receiving multicast data, it is possible to have all the interfaces on a computer join a multicast group. But for the purposes of outputting multicast data, only one interface should join the group.

sACN does allow for broadcast or unicast but this functionality is not widely used.

KiNET

KiNET is intended to be unicast only. Therefore all packets are addressed to specific IP addresses. This is efficient, but as with Art-Net unicast, the user must know all of the IP addresses of the connected devices that they want to communicate with. Also, the Mbox computer must still have a port that is on the same subnet as the receiving device.

Notes

B.

PARAMETER MAPPING

This Appendix contains tables for the combined parameter map, as well as, additional controls such as playmodes, tiling, shutter shapes, aspect control, blending control, texture effect, transitions, and built-in shapes.

+   PARAMETER MAPPING CHARTS

Parameter Mapping Charts

One Mbox server running v3.6 software or greater is made up of between 14 and 22 fixtures depending on the operating mode. For ease of use, some consoles may split the Layer fixture into two separate fixtures. Previous operating modes have significant differences in the arrangement of parameters. In all cases, a show programmed to run in one particular operating mode will not be compatible with a server running another mode. This configuration includes lighting, camera, shutter, keystone, and object controls.

Note: All 16-bit values are in Big Endian format. For example, the value 23 would be presented as 0 in the first channel and 23 in the second channel.

For additional instructions on parameter mapping, refer to the following online videos:

Layer, Frame Blending, Play Modes: Mbox Media Server Training Video Chapter 11: Play Modes

Transitions: Mbox Media Server Training Video Chapter 13: Transitions

Texture Effects: Mbox Media Server Training Video Chapter 17: Texture Effects

Summary: Single Output

Start	Size	Type	Universe
1	11	Master	
12	8	Keystone	
20	14	Shutter	1st
34	39	Layer 1	
73	39	Layer 2	190 Total Channels
112	39	Layer 3	
151	39	Layer 4	

Summary: Dual Output - Panoramic Wide

Start	Size	Type	Universe
1	11	Master	
12	8	Keystone	
20	14	Shutter	1st
34	39	Layer 1	
73	39	Layer 2	190 Total Channels
112	39	Layer 3	
151	39	Layer 4	

Parameter Descriptions

Fixture	Parameter	Description
Master	Pixel Map Level	Intensity master for pixel mapping Art-Net output
	Control Selector	Set in combination with the Master Control channel for macro selection
	Texture Effect 1	2D effect applied to the final image
	Modifier 1a	First modifier
	Modifier 1b	Second modifier
	Red	Red subtractive
	Green	Green subtractive
	Blue	Blue subtractive
	Intensity	Intensity of the video output
	Control	Control functions for server - HUDs, macros, etc.
	Master Audio Volume	Final output volume master
Keystone	Keystone 1a	Bottom-Left X or X Rotation Coarse
	Keystone 1b	Bottom-Left Y or X Rotation Fine
	Keystone 2a	Bottom-Right X or Y Rotation Coarse
	Keystone 2b	Bottom-Right Y or Y Rotation Fine
	Keystone 3a	Top-Left X or Aspect Coarse
	Keystone 3b	Top-Left Y or Aspect Fine
	Keystone 4a	Top-Right X or Curve Y
	Keystone 4b	Top-Right Y or Curve X

Parameter Descriptions (Continued)

Fixture	Parameter	Description
Shutter	Red	Red additive
	Green	Green additive
	Blue	Blue additive
	Scale	Adjusts the size of the shutters
	Edge Softness	Adjusts the soft edge of the shutters
	Shutter 1a	Bottom-left corner L/R adjustment
	Shutter 1b	Top-Left corner L/R adjustment
	Shutter 2a	Top-Left corner U/D adjustment
	Shutter 2b	Top-Right corner U/D adjustment
	Shutter 3a	Top-Right corner L/R adjustment
	Shutter 3b	Bottom-Right corner L/R adjustment
	Shutter 4a	Bottom-Right corner U/D adjustment
	Shutter 4b	Bottom-Left corner U/D adjustment
Layer	Effect 1	2D or 3D effect applied to the layer
	Modifier 1a	First modifier
	Modifier 1b	Second modifier
	Blend Mode	Adjusts how layer blends with layers underneath
	Draw Mode	Adjusts methods for drawing the layer
	Red	Red subtractive
	Green	Green subtractive
	Blue	Blue subtractive
	Opacity	Transparency of the layer
	Texture Folder	Selects numbered content folder from w/in Media folder

Parameter Descriptions (Continued)

Fixture	Parameter	Description
Layer (continued)	Texture File	Selects numbered file from within Texture folder
	Play Mode	Selects play mode of movie content
	Play Speed	Adjusts play speed of movie content
	In Frame	Sets in-frame of movie content
	Out Frame	Sets out-frame of movie content
	Sync Stream	Selects 1 of 16 channels for playback synchronization
	Sync Offset	Adjusts layer's playback sync offset later/earlier
	Frame Blending	Adjusts the amount of interframe blending for movie content
	Texture X-Fade Type	Selects the type of crossfade for 2D content on the layer
	Texture X-Fade Time	Adjusts the timing of 2D crossfades on the layer
	X Position	Moves the layer left and right
	Y Position	Moves the layer up and down
	Scale	Adjusts the size of the layer
	X Scale	Adjusts the width of content on the layer
	Y Scale	Adjusts the height of content on the layer
	Z Rotation	Adjusts rotation of the layer through the Z axis
	X Rotation	Adjusts rotation of the layer through the X axis
	Y Rotation	Adjusts rotation of the layer through the Y axis
	Object File	Selects numbered file from folder 000 within Models folder

Master

Chan	Size	Function	Default	Snap
1	1	Pixel Map Level	255	N
2	1	Control Selector	0	Y
3	1	Texture Effect 1	0	Y
4	1	Modifier 1a	0	N
5	1	Modifier 1b	0	N
6	1	Red	255	N
7	1	Green	255	N
8	1	Blue	255	N
9	1	Intensity	255	N
10	1	Master Control Channel	0	Y
11	1	Master Audio Volume	255	N

Master Control Channel

Values	Command
0	None/Idle
10-19	Version HUD
20-29	Performance HUD
40-49	Master/Keystone/Shutter HUD
50-59	Layer HUD 1
70-71	Raw Art-Net HUD
80-84	Show pix-map context view
85-89	Show pix-map fixture view
110-111	Timecode HUD Center
112-113	Timecode HUD Top Right
114-115	Timecode HUD Top Left
116-117	Timecode HUD Bottom Right
118-119	Timecode HUD Bottom Left
120	Show Alignment Rectangles No Group, No X or O
121	Show Alignment Rectangles Group 1, No X or O
122	Show Alignment Rectangles Group 2, No X or O
123	Show Alignment Rectangles Group 3, No X or O
124	Show Alignment Rectangles Group 4, No X or O
125	Show Alignment Rectangles Group 5, No X or O
126	Show Alignment Rectangles Group 6, No X or O
127	Show Alignment Rectangles Group 7, No X or O
128	Show Alignment Rectangles Group 8, No X or O
129	Show Alignment Rectangles Group 9, No X or O
130	Show Alignment Rectangles Group, No X and O
131	Show Alignment Rectangles Group 1, X and O

Master Control Channel (Continued)

Values	Command
132	Show Alignment Rectangles Group 2, X and O
133	Show Alignment Rectangles Group 3, X and O
134	Show Alignment Rectangles Group 4, X and O
135	Show Alignment Rectangles Group 5, X and O
136	Show Alignment Rectangles Group 6, X and O
137	Show Alignment Rectangles Group 7, X and O
138	Show Alignment Rectangles Group 8, X and O
139	Show Alignment Rectangles Group 9, X and O
201	Run script (script selected by index value set on Master Control Selector parameter) **
202	Switch to Pixel Mapping file (file selected by index value set on Master Control Selector parameter) **
203	Switch to Alignment Rectangle file (file selected by index value set on Master Control Selector parameter) **
220	Cancel keyboard HUD
230	File Sharing On
231	File Sharing Off
232	ARD On
233	ARD Off
234	Become Backup (turn on Backup mode pref)
235	Become Primary (turn off Backup mode pref)
240	Rescan Media Library *
244	Enable Pixel Mapping Output *
245	Disable Pixel Mapping Output *
246	Enable Masked Pixel Mapping fixtures *
247	Disable Masked Pixel Mapping fixtures *
250	Quit Mbox Application *
251	Shutdown Computer *
252	Restart Computer *
253	Restart Mbox Application *

*	Special commands require that the parameter value be held for three seconds, followed by an immediate change to value 0 (without any other values in between).

**	Special command that requires the value to be held for 250mS, followed by an immediate change to 0 (with no values in between)

Keystone

Chan	Size	Function	Default	Snap
1	1	Corner 1-X	0	N
2	1	Corner 1-Y	0	N
3	1	Corner 2-X	0	N
4	1	Corner 2-Y	0	N
5	1	Corner 3-X	0	N
6	1	Corner 3-Y	0	N
7	1	Corner 4-X	0	N
8	1	Corner 4-Y	0	N

Note: Keystone, Warp, and Blend settings created in the Mbox Remote application will override any Art-Net controlled keystone values.

Shutter

Chan	Size	Function	Default	Snap
1	1	Red	0	N
2	1	Green	0	N
3	1	Blue	0	N
4	2	Scale	32767	N
6	1	Edge	0	N
7	1	Shutter 1a	0	N
8	1	Shutter 1b	0	N
9	1	Shutter 2a	0	N
10	1	Shutter 2b	0	N
11	1	Shutter 3a	0	N
12	1	Shutter 3b	0	N
13	1	Shutter 4a	0	N
14	1	Shutter 4b	0	N

Layer

Chan	Size	Function	Default	Snap
1	1	Effect 1	0	Y
2	1	Effect Control 1a	0	N
3	1	Effect Control 1b	0	N
4	1	Layer Blend Mode	0	Y
5	1	Draw Mode	0	Y
6	1	Red	255	N
7	1	Green	255	N
8	1	Blue	255	N
9	1	Opacity	0	N
10	1	Texture Folder	0	Y
11	1	Texture	0	Y
12	1	Play Mode	0	Y
13	1	Play Speed	127	N
14	2	In Frame	0	N
16	2	Out Frame	65535	N
18	1	Sync Stream	0	Y
19	1	Sync Offset	127	N
20	1	Frame Blending	255	N
21	1	Texture Xfade Type	0	Y
22	1	Texture Xfade Time	0	Y
23	2	X Position	32767	N
25	2	Y Position	32767	N
27	2	Scale	32767	N
29	2	X Scale	32767	N
31	2	Y Scale	32767	N
33	2	Z Rotation	32767	N
35	2	X Rotation	32767	N
37	2	Y Rotation	32767	N
39	1	Object	0	Y

Layer Blend Mode

Values	Mode	Description
0	Default Overlay	no layer blending
1	Additive	layer's colors are added to underlying colors (blacks appear transparent)
2	Screen	similar to above, with less of underlying color (blacks appear transparent)
3	Multiply	multiplies layer's color with underlying color (blacks appear opaque)
4	Subtractive	layer's colors are subtracted from underlying colors (blacks appear transparent)
5	Exclusion	underlying colors are inverted where layer color is lighter; layer's colors are then added to underlying colors (blacks appear transparent)
6	Invert Subtractive	underlying colors are subtracted from layer's colors (blacks appear opaque)
7	Invert Additive	underlying colors are added from layer's colors (blacks appear transparent)

Draw Mode

Values	Mode	Description
0	NONE	no effect
1	Light/Trim	automatic ambient lighting of object/trims edge of backgrounds
2	Cut	cuts holes in stencil mask
3	Cut + Light	as above w/ lighting
4	Cut & Draw	as mode 2 but texture is drawn on object too
5	Cut & Draw + Light	as above w/ lighting
6	Draw thru Stencil	this layer's texture is drawn where holes have been cut in stencil
7	Draw thru Stencil + Light	as above w/ lighting
8	Draw onto Stencil	this layer's texture is drawn where stencil is not cut
9	Draw onto Stencil + Light	as above w/ lighting

Play Mode

Values	Play Mode
0	Forward Loop
1	Forward Loop, Pause when Layer Opacity = 0
2	Forward Loop, Pause and Reset to In-Point when Layer Opacity = 0
10	Reverse Loop
11	Reverse Loop, Pause when Layer Opacity = 0
12	Reverse Loop, Pause and Reset to In-Point when Layer Opacity = 0
20	Forward Once
21	Forward Once, Pause when Layer Opacity = 0
22	Forward Once, Pause and Reset to In-Point when Layer Opacity = 0
30	Reverse Once
31	Reverse Once, Pause when Layer Opacity = 0
32	Reverse Once, Pause and Reset to In-Point when Layer Opacity = 0
40-49	Forward Bounce
50-59	Reverse Bounce
60-69	Random
80-89	Scrub - In Frame
90-99	Scrub - Out Frame
130-134	Timecode Sync (Strict Lock)
135	Timecode Sync then Freewheel
136	Timecode Jam Sync
140 - 149	Layer Slave
150 - 159	Layer Master - Forward Loop
160 - 169	Layer Master - Forward Once
180	Kiosk Mode Non-Looping, pause on last frame of last file
181	Kiosk Mode Non-Looping -loop last file
182	Kiosk Mode Non-Looping - fade out last file
185 - 189	Kiosk Mode Looping
190 - 199	SMPTE Kiosk Mode
250-255	Restart Movie from In Point

Sync Stream

Values	Mode
0	Layer to Layer Stream (default)
1-2	Stream Number

Frame Blending

Values	Description
0	No Frame Blending (more accurately, blend time = 0)
1-255	Variable Frame Blend Time, as a proportion of the frame time. This is a square-law control, and 50% blend time is achieved at value 210, 25% blend time at value 165.

Video Input and Utility (Texture Folder 255)

Texture Value	Input
0	Patch Info Display
121	Layer Copy FX+ Layer 1
122	Layer Copy FX+ Layer 2
123	Layer Copy FX+ Layer 3
124	Layer Copy FX+ Layer 4
201	Syphon Input 1
202	Syphon Input 2
240	CG Color bars
254	Video Input 2
255	Video Input 1

TC DMX Control

The TC DMX controls are only required when the TC DMX Control feature has been configured. (Refer to) Each item in the TC DMX Controls preference relates to one fixture that must be patched to control timecode on a layer.

Chan	Size	Function	Default	Snap
1	1	Hours (0-23)	255	N
2	1	Minutes (0-59)	255	N
3	1	Seconds (0-59)	255	N
4	1	Frames (0-29)	255	N

Effects

Refer to notes at end of Effects table.

ID	Effect	Description	Control 1	Control 2
0	NONE	no effect		
1	Monochrome	convert colors to grayscale	amount	
2	Sepia Tone	convert colors to sepia tone image	amount	
3	Invert	color invert		
4	Color Reduce	reduce color space	amount	
5	Bright/Cont	brightness and contrast controls	brightness	contrast
6	Saturation	saturation control	saturation	
7	Hue	adjustment hue	hue	
8	Gamma	gamma adjustment	gamma	
9	Exposure	exposure adjustment	exposure	
10	Solarize	solarize effect	amount	
11	Black&White	convert image to black/white	threshold	
12	Black Alpha	dark areas transparent	threshold	
13	X-Ray	inverted grayscale	amount	
14	Bloom	soften edges, add glow	radius	intensity
15	Gloom	dulls highlights	radius	intensity
16	Sharpen	increases image detail by sharpening	sharpness	
17	Median	reduce noise with median calculation		
18	Color Switch	RGB->RBG/BGR/BRG/GBR/GRB	mode	intensity
19	Vertical Lines	spread single row over vertical space	row	
20	Color Shift	dynamic color shift (sine function)	range	speed
21	Blur	simple/quick image blur	amount	
22	Gaussian Blur	more sophisticated/slow blur	amount	
23	Circular Screen	circular shaped halftone screen	width	sharpness
24	Dot Screen	dot patterned halftone screen	width	angle
25	Line Screen	line patterned halftone screen	width	angle
26	Hatched Screen	hatch patterned halftone screen	width	angle
27	CMYK Halftone	color, halftoned rendition	width	angle
28	Tiles	image tiling	divisions	
29	Circular Crop	circular image crop with edge blur	size	edge
30	Rectangular Crop	rectangular image crop with edge blur	size	edge
31	Mirrors	various mirror modes	mode	
32	Horizontal Bars	break image into bars	number	width
33	Vertical Bars	break image into bars	number	width
34	Double Vision	offset image and overlay	x offset	y offset
35	Circular Crop w/ X/Y Position	Crop that can be repositioned [1]	size	softness
36	Rectangular Crop w/ X/Y Pos.	Crop that can be repositioned [1]	size	softness
37	Rippling	simple ripple effect	size	granularity
38	Flicker	dynamic flickering effect	size	speed
39	Shake	dynamic shaking effect	size	speed
40	Wobble	dynamic wobbling effect	size	speed
41	Bump Distortion	bump distortion	radius	scale
42	Hole	hole distortion	radius	

Effects (Continued)
Refer to notes at end of Effects table.

ID	Effect	Description	Control 1	Control 2
43	Pinch	pinch distortion	radius	scale
44	Torus	torus distortion	width	thickness
45	Twirl	twirl distortion	radius	angle
46	Vortex	vortex distortion	radius	angle
47	Lozenge	lozenge distortion	width	thickness
48	Edge Work	B&W conversion	radius	
49	Edge Detect	edge detection with color	intensity	
50	Crystallize	break up into crystal pattern	radius	
51	Pixelate	pixelates image, square	scale	
52	Pointillize	break image into points	radius	
53	Kaleidoscope	geometric distortion	angle	divisions
54	LED Wall	break up into dots	dot amount	dot size
55	Cartoon	cartoon effect	line width	color reduction
56	White Alpha	white areas transparent	threshold	
57	Red Alpha	red areas transparent	threshold	
58	Green Alpha	green areas transparent	threshold	
59	Blue Alpha	blue areas transparent	threshold	
60	White Only	everything but white areas transparent	threshold	
61	Red Only	everything but red areas transparent	threshold	
62	Green Only	everything but green areas transparent	threshold	
63	Blue Only	everything but blue areas transparent	threshold	
64	Zoom Blur	blurs from center of image	size	
65	Unsharp Mask	increases image detail by sharpening	intensity	radius
66	Motion Blur	blurs along a variable axis	intensity	direction
67	Op Tile	glass block tile effect	scale	tile size
68	Circular Wrap	wraps image into tube shape	size	rotation
69	Circular Splash	clamps image from center outwards	size	
70	Orth Shutter	horizontal & vertical 90° shutters	H insertion	V insertion
71	Horiz Crop	horizontal 90° shutters	insertion	center (127=def.)
72	Vert Crop	vertical 90° shutters	insertion	center (127=def.)
73	Vertical Roll	vertical roll	speed	pause
74	Vertical Roll Up	vertical roll up	speed	pause
75	Horizontal Roll	horizontal roll	speed	pause
76	Horizontal Roll Left	horizontal roll left	speed	pause
77	Black Alpha 2	revised black alpha effect	intensity	threshold
78	Black Threshold	renders black areas as true black	intensity	threshold
79	Pos Lattice	divides image into rectangles	divisions	size
80	Neg Lattice	reverse of pos lattice	divisions	size
81	Simple Duotone	two-color duotone effect	mode	threshold
82	Channel Shift	separates RGB channels	horizontal	vertical
83	LumaKey	renders bright areas transparent	intensity	threshold
84	LumaKey Inverse	renders dark areas transparent	intensity	threshold
85	Comic Effect	posterizing/halftone effect		
86	Hexagonal Pixelate	pixelates image, hexagonal	scale	

Effects (Continued)
Refer to notes at end of Effects table.

ID	Effect	Description	Control 1	Control 2
87	Glide Reflected Tile	rectangular tile effect	rotation	size
88	4-fold Rotated Tile	four-sided tile effect	rotation	size
89	6-fold Rotated Tile	six-sided tile effect	rotation	size
90	6-fold Reflected Tile	six-sided tile effect	rotation	size
91	12-fold Reflected	twelve-sided tile effect	rotation	size
92	Layer Edge Blend Left	soft edge to transparent	amount	softness
93	Layer Edge Blend Right	soft edge to transparent	amount	softness
94	Layer Edge Blend Top	soft edge to transparent	amount	softness
95	Layer Edge Blend Bottom	soft edge to transparent	amount	softness
96	Layer Edge Blend Left & Right	soft edge to transparent	amount	softness
97	Layer Edge Blend Top & Bottom	soft edge to transparent	amount	softness
98	Alpha Mask From File *	mask using additional file	file number	Flip mode
100	Drop Shadow 2	simple drop shadow	offset	amount
101	Luma Lines	color & luma block effect	width	rows
102	ASCII Art	converts image to characters	scale	desaturate
108	Shaded Material *	additional file creates relief map	file number	amount
109	Hue and Saturation	combines hue and saturation controls in a single effect	Hue (127 = default)	Saturation (127 = default)
110	Matte from Layer	Creates alpha matte using selected layer	layer number	mode [2]
111	Decay	Creates decay trails	amount	n/a
112	Meta Image [3]	pixelates image by applying substitute image based on luma calculation	scale	desaturate
201	L/R-Shutter [4]	mask LHS/RHS of layer	left	right
202	B/T-Shutter	mask Bot/Top of layer	bottom	top
203	Move Center	shift rotational center of image	X (127=def.)	Y (127=def.)
204	Shake	X/Y shake effect	X	Y
205	Strobe	strobe effect	off time	on time
206	Object Tile	tiling effect	number	spacing
207	Z-Position	changes z-position of object	position	
208	Specular Highlight	creates specular highlight from spotlights	amount	spread
209	Texture Scale/Rotation	adjusts scale & rotation of texture on gobos	scale	rotation
210	Texture X/Y Position	adjusts X/Y position of texture on gobos	X	Y
211	n/a			
212	Crop to Alignment Rectangle	crops content to screen area defined by alignment rectangle	alignment rectangle number	1>63 no action, 64>127 center content
213	X/Y Position and Scale Damping	motion damping for position & scale **	X/Y position	scale
214	Z Rotation and Scale Damping	motion damping for position & scale **	Z rotation	scale

Effects (Continued)
Refer to notes at end of Effects table.

ID	Effect	Description	Control 1	Control 2
215	Texture Flip	inverts drawing axes (X,Y,Z) when applying a texture to an object	**Control 1:** 0 - 31 = flip X 32 - 63 = no flip 64 - 95 = flip X, Y 96 - 127 = flip Y 128 - 159 = flip X, Z 160 - 191 = flip Z 192 - 223 = flip X, Y, Z 224 - 255 = flip Y, Z **Control 2: N/A**	
217	Spin	controls spin speed and direction about X/Y/Z axes	**Control 1 - Z Axis :** 0 = stop + return to 0° 1 - 126 = fast to slow forwards 127 = stop 128 - 255 = slow to fast backwards **Control 2 - X Axis :** 0 = stop + return to 0° 1 - 126 = fast to slow forwards 127 = stop 128 - 255 = slow to fast backwards **Control 3 - Y Axis :** 0 = stop + return to 0° 1 - 126 = fast to slow forwards 127 = stop 128 - 255 = slow to fast backwards **Control 4 - N/A**	

* These elements require additional input files in order to obtain the desired result. Users will need to provide their own additional files.

** The modifier channels for the Damping effects apply damping in 1/30 sec. increments. Therefore, a value of 30 will add 1 second to any change of that parameter. The damping is divided equally between the start and end of the movement curve. Effect parameters can be adjusted during a move to affect only one side of the curve.

Notes for Effects:

[1] These effects require that the second effect channel on the corresponding layer be set to 255. At that point, Mod 2A controls X-position and Mod 2B controls Y-position. The default value for both additional controls is 127. (These effects will not work properly with Mbox Mini, as it only has one effect per layer.)

[2] Modes for Effect 110:
0 Luma, 1 Luma Invert, 2 Red, 3 Red Inv, 4 Green, 5 Green Inv, 6 Blue, 7 Blue Inv, 8 Alpha, 9 Alpha Inv

[3] The Meta Image should be placed in the */Mbox/plugins/images/masks/* folder and be named as "011.filename.file extension." File can be JPEG or PNG (transparency not supported).

[4] Effects 201-217 are not available for use on the Master fixture.

Transitions

Refer to notes at end of Transitions table.

ID	Transition	Description
0	Dissolve	dissolve
1	Dissolve2	dissolve 2
2	Wipe Right	slightly blended right to left wipe
3	Wipe Left	slightly blended left to right wipe
4	Wipe Down	slightly blended top to bottom wipe
5	Wipe Up	slightly blended bottom to top wipe
6	Wipe Diagonal	slightly blended diagonal wipe
7	Wash Right	more blended right to left
8	Wash Left	more blended left to right
9	Wash Down	more blended top to bottom
10	Wash Up	more blended bottom to top
11	Wash Diagonal	more blended diagonal
12	White Right	blended white stripe left to right
13	White Left	blended white stripe right to left
14	White Down	blended white stripe top to bottom
15	White Up	blended white stripe bottom to top
16	White Diagonal	blended white stripe diagonal
17	Through Black	fade to black and back in
18	Through White	fade to white and back in
19	Through Red	fade to red and back in
20	Bright First	transition by brightness of new image
21	Dark First	transition by darkness of new image
22	Dots	fade with small dots
23	Big Dots	fade with big dots
24	Burst	white star burst transition
25	Flash	white flash transition
26	Slow Dissolve	dissolve that comes in more slowly
27	Slower Dissolve	dissolve that comes in even more slowly
28	Rotate Left	simple 3d rotate effect
29	Rotate Right	simple 3d rotate effect
30	Rotate Down	simple 3d rotate effect
31	Rotate Up	simple 3d rotate effect
32	Rotate Center Vertical	simple 3d rotate effect
33	Rotate Center Horizontal	simple 3d rotate effect
34	Zoom Out	zoom out and back in
35	Zoom In	zoom in and back out
36	Horizontal Bars	10 horizontal bars, hard edge
37	Horizontal Bars + Blend	10 horizontal bars, blended edge
38	Vertical Bars	20 vertical bars, hard edge
39	Vertical Bars + Blend	20 vertical bars, blended edge
40	Circle Center	circle out from center
41	Circle Center + Blend	circle out from center, blended edge
42	Concentric Circles	circles out from center, blended edges

Transitions (Continued)
Refer to notes at end of Transitions table.

ID	Transition	Description
43	Push Right	new image pushes in left to right
44	Push Left	new image pushes in right to left
45	Push Down	new image pushes in top to bottom
46	Push Up	new image pushes in bottom to top
47	Split Right	old image splits and slides to the right
48	Split L/R Center	old image splits and slides left and right from the center
49	Split Down	old image splits and slides down
50	Split Up	old image splits and slides up
51	Split U/D Center	old image splits and slides up/down from the center
52	Split XY	old image splits slides out from center
53	Bar Swipe Right	old image slides to right in strips
54	Bar Swipe Left	old image slides to left in strips
55	Bar Swipe Up	old image slides upward in strips
56	Bar Swipe Down	old image slides downward in strips
57	Page Curl 1 *	curls from the bottom-right corner of the image to the top-left corner.
58	Page Curl 2 *	curls from the top-right corner of the image to the bottom-left corner.
101-110	Custom Hard Wipe 1-10 **	hard edge wipe generated from additional image file
111-120	Custom Soft Wipe 1-10 **	soft edge wipe generated from additional image file
255	Object Fade	fade out on current object, fade in on new object

* The Page Curl transitions attempt to use files 021 and 022 in the *Mbox/plugins/images/grayscale* folder. If these files do not exist, the transitions will not work properly. File 021 is a 1920 x 1080 50% gray, and file 022 is a 480 x 480 radial gradient. Both files are saved as JPEGs.

** These elements require additional input files in order to obtain the desired result. Users will need to provide their own additional files.

Mbox® Mini User Manual

Version as of: December 14, 2014

PRG part number: 02.9800.0004.39

Production Resource Group, LLC
Dallas Office
8617 Ambassador Row, Suite 120
Dallas, Texas 75247
www.prg.com