Creating Custom 3D Content for

August 3, 2009

Contents

Introduction

This document provides details on creating custom 3D content for use with of Instant Effects Presenter. It is part of the Instant Effects Authoring Toolkit (Authoring Toolkit). The Authoring Toolkit contains plug-ins and utility programs that enable Instant Effects customers who are familiar with 3D authoring to add new FXThemes™ (themes) and Insert Scenes and custom Transitions to their Instant Effects presentations.

Notes:

1. In order to utilize this Toolkit effectively it is necessary for the reader to have a full understanding of the Instant Effects product line and all the latest capabilities of the Presenter software.
2. Actual 3D content creation requires 3ds max® modeling and animation software from Autodesk.
3. A powerful graphics computer is required to run Presenter capably. This is especially true with some of the new features of Presenter 4.0. Make sure to test custom 3D assets created with this Authoring Toolkit on the systems where they will be deployed.

Instant Effects Presenter Product Line

With the latest release of Presenter the Instant Effects product line has been expanded. Where previously there was a single product named OfficeFX Presenter, there are now three additional specialty versions. The Instant Effects product line now consists of:

- **OfficeFX Presenter**: The same Presenter capabilities as before but limited to 1920x1200 resolution and a single file (vs. live) video display on any one slide. OfficeFX Presenter is configured for the requirements of digital media specialists, smaller event production companies, and creative services departments within companies large enough to employ such specialists
- **Event Presenter**: All Presenter features plus
  - Ability to run at any output resolution and aspect ratio.
  - Ability to add two videos to any slide
  - Support for live video
  - Ability so send single wide display to 2 or 3 projectors with overlapped and blended output

  Event Presenter is configured for the needs of event production specialty firms that require a cost effective solution for producing high end shows with multiple video inputs and at potentially ultra wide aspect ratios.
- **S3D Presenter**: All OfficeFX Presenter features plus the ability to generate displays in a variety of standard Stereo 3D (S3D) formats.
- **Master Presenter**: Support for combined feature set of all the above products.

Notes:

1. Within this and other Instant Effects documentation we use often shorten the product name to either Presenter.
2. Unless specifically stated, the instructions in this Authoring Toolkit apply to all the above products.
3. The name OfficeFX® is a registered trademark of Instant Effects. It is baked into the Presenter file structure. You will see it frequently as we go through the creation of new Presenter 3D assets.

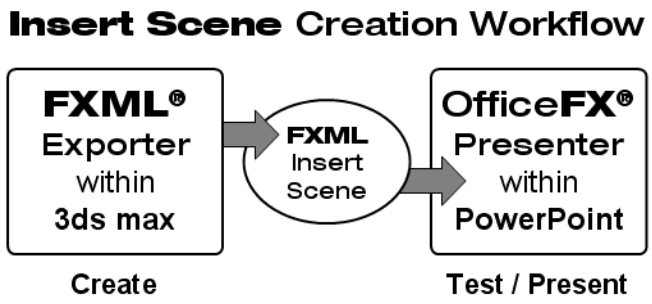
Customizing Presenter

The Instant Effects Presenter product line is an “open system”. It is designed specifically for professional digital artists who create and deliver custom presentations for others. Central to the focus of this document – Presenter offers full customization capability for all aspects of a presentation. The following 3D elements of an Instant Effects presentation can be customized with Presenter.

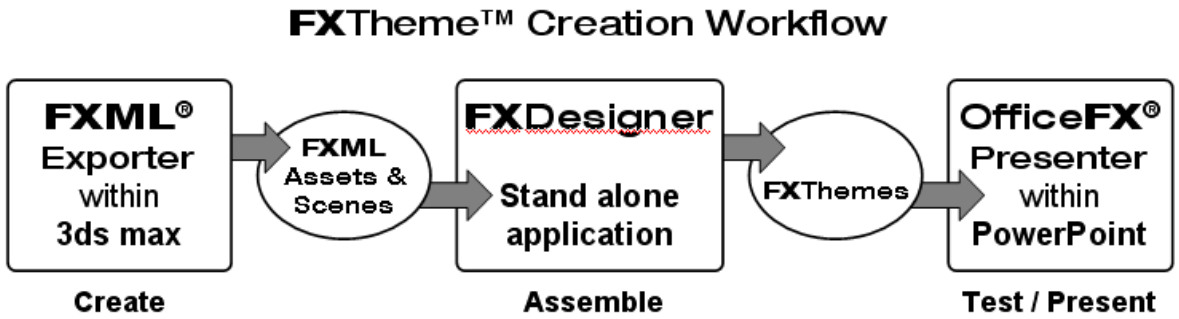
- **Insert Scenes**: 3D scenes that can be added to any Instant Effects presentation. Insert Scenes are commonly used to represent company products and logos.
- **FXThemes**: The visual “templates” within Presenter that control the look and behavior of a user’s PowerPoint content. Custom FXThemes are commonly developed to amplify the brands and trademarks companies that use Presenter.
- **Video Insert Scene**s: Are Insert Scenes that allow arbitrary (user selected) video to be streamed onto some portion of the scene geometry.
- **FXTransition Packs**: Are custom packs of transitions that can be applied to any Instant Effects presentation.

Custom 3D Content Creation for Instant Effects Presenter

The process of building custom 3D content for Presenter will vary according what form of content is being created. For example, creating Insert Scenes is accomplished with simple export from the 3ds max®, modeling and animation system from Autodesk. See below.



In contrast, building a complete FXTheme with multiple Variations and Transitions is a more complex task that requires both the FXDesigner application and a deeper understanding of the real time scene compositing techniques employed within the Instant Effects product architecture. The process includes the creation of scenes and asset libraries using 3ds max and the FXML Exporter, then the organization of those scenes and assets into finished FXThemes and FXTheme Packs using the Instant Effects FXDesigner™. See below.



The majority of this document is dedicated to the details of FXTheme creation. In it we provide definitions for FXTheme terminology, detail theme FXTheme structural concepts, and make recommendations on theme creation techniques and data organization. It is imperative that the reader have an understanding of both the Presenter product and functional range of FXThemes before working through this document and the theme creation process. If the goal is building finished FXThemes, the FXTheme Structure and Concepts sections should be reviewed before moving to the Tutorials section.

Creation of Insert Scenes is significantly less complex. Users looking only to create Insert Scenes should read the Definitions section that follows, then skip forward to the FXML Export Utility section, and finally move to the Tutorial section of where Insert Scenes are the first topic covered.

Tools

The following tools are required for creating custom 3D content for use with Instant Effects Presenter:

- **3ds max from Autodesk** (versions 7, 8, 9, 2008, 2009, and 2010 are supported). . These tools serve as the source for 3D scenes, geometry, materials, lighting, and animation.
- **FXML Exporter** utility plug-ins for any of the above supported 3ds max versions for converting 3D content into FXML™ scenes, assets, and asset libraries. These plug-ins are included with the Authoring Toolkit.
- **FXDesigner for PowerPoint** stand alone application for defining FXThemes and knitting together the FXML scenes and assets created by the FXML Exporter. The FXDesigner is included with the Authoring Toolkit.
- **PowerPoint 2002**, **PowerPoint 2003, or PowerPoint 2007** from Microsoft.

The Instant Effects FXML Exporter utility takes elements of 3ds max files and exports them as FXML format scene and asset files. In the case of Insert Scenes, those scenes are immediately ready for use in Presenter. When the goal is creation of either FXThemes or FXTransition Packs, the exported scenes are building blocks that will contribute to the final result. The Instant Effects FXDesigner is a stand alone application that assembles those FXML building blocks into finished FXThemes and FXTransition Packs. FXDesigner runs only alongside Presenter.

Definitions

The following are important elements of the FXTheme building process. Definitions are provided to establish a common understanding and establish a clear notation for documenting that process.

FXML®

FXML stands for “Special Effects Markup Language”. FXML is the underlying file format of Instant Effects. FXThemes and their components are built up from files in the FXML format.

FXML Exporter

This is the utility that operates from within a 3D authoring package that exports content from that tool into the FXML format. Instant Effects is working to establish FXML export solutions from primary 3D authoring tools available in the market today. The first of those is 3ds max from Autodesk. This document assumes the FXML Exporter in use is the exporter from 3ds max.

FXThemes

FXThemes are the complete definitions that Presenter uses to determine the look and behavior of a user’s PowerPoint content. FXThemes can contain multiple foreground and background 3D scenes as well as scene description assets that control appearance and animation of PowerPoint text and graphics elements.

Theme Packs

Theme Packs are collections of FXThemes that can share common assets. The Theme Pack concept is implemented to facilitate the addition of new and custom FXThemes that may serve a common purpose. For example, companies using Presenter may build packs of themes that serve different purposes, but share logos, brands, and products.

FXML Scenes

FXML scenes contain all the vital elements of a complete 3D scene. This includes a camera, lighting, geometry, material definitions, and optionally – animation. FXML scenes are commonly used as the background and foreground design elements of FXThemes. Special "Insert" FXML scenes can also be defined to allow addition of 3D objects to any Instant Effects presentation.

Clips

Clips are animation assets that make Presenter interactive. They are an important element of FXThemes and may be a new concept for many 3D artists who are most familiar with creating purely linear media. Both PowerPoint and Presenter are non-linear and interactive applications. How and when their content is displayed can be controlled by user input at any time during a presentation. The simplest example is the triggering of a transition from slide A to slide B. Clips are the Presenter animations that play during those user triggered events. They are specially tagged within 3ds max so that Presenter recognizes them properly.

Variations

FXThemes can contain multiple different looks and settings. These are termed "variations". In some ways, Presenter's variations are similar to PowerPoint's "Layouts". They are often set up to accommodate a particular type of slide – e.g. a title slide vs. a content slide. While only one FXTheme can be applied to a PowerPoint file at a time, a broad variety in appearance and behavior within the theme can be achieved through variations. Different variations from an FXTheme can be applied to each slide in a presentation.

Transitions

Transitions are effects that occur between slides. When set to do so, Presenter will respect the PowerPoint transitions that are selected by the user. In general, Presenter users employ the Custom Transitions that come with the product. Those transitions are created in FXDesigner by referencing FXML scenes and assets as part of the FXTheme development process.

FXML Assets

FXML assets are scene components that are stored in FXML format. FXML assets are comprised mostly of libraries (files) containing motion and material definitions. FXML assets are used by an FXTheme to determine the look and behavior of the user's PowerPoint content when it is displayed in Presenter. The scene displaying PowerPoint content is constructed by Presenter at run time.

Asset File names

- **Material.fxml** – Material library for a theme's content (Titles, Bulleted Text, Lines, etc)
- **MaterialClip.fxml** – Animated Material Library for a theme's content for use in interactive functions such as bullet text selection or individual content transitions
- **FloatClip.fxml** – Animated visibility curves for making content appear and disappear.
- **Transitions.fxml** – Stores settings for a theme's individual custom transitions
- **TransformClip.fxml** – Library of animation for use with content selection and transitions
- **Transform.fxml** – Library of animation for use with the 3D bullets
- **ThemeWizSettings.xml** – Default "start-up" settings for the theme

Asset File Types

- **<Theme name>.FXML** – this is the theme's default variation file
- **<Theme Name>.<Variation Name>.FXML** - secondary variation files for the theme
- **SC__<scene name>.FXML** - Scenes created by the FXML Exporter Utility are prefixed with SC__. Those are used for the theme's backgrounds, foregrounds, and transitions.
- **<object name>.X** – 3D geometry is exported with the .x suffix.

FXDesigner

The FXDesigner is a stand alone application that assembles all FXML scenes and assets together into a finished FXTheme. It does not create any new scenes or assets. It does write FXTheme, FXTheme variation files, and transition files. Those files are also in the FXML format.

FXTheme Structure and Concepts

FXThemes can impart a wide variety of looks and behaviors to PowerPoint content. There are a number of ways in which they can be structured. Understanding those options is important if the full range of creative options is to be accessed. In this section we introduce basic concepts of FXTheme structure. This includes both how an FXTheme is arranged on disk and the way its 3D assets and scenes are layered for display within themes.

FXTheme File Structure

Only a single FXTheme can be used at any one time by Presenter, meaning any individual PowerPoint file can only contain instructions for one FXTheme. When you first select a theme, its default settings are loaded into Preferences, Slide Options and Variations/Transitions. If you switch themes during an OFX session, the slides are set to the new theme's "Default" variation and transition. While the FXTheme description is contained within a single FXML file folder it may reference other external files. Where these other external files are placed impacts whether they can be shared between themes in a Theme Pack, or will be only available to a particular FXTheme. Themes are organized on disk in the following hierarchy:

System\Default Level

The *\OfficeFX\Repository* folder contains assets that are used by Presenter as default settings for presentation attributes when a theme does not define those attributes. These defaults allow a theme to function without being completely defined. The system level assets should not be modified because they will impact themes from all other sources.

Theme Pack level

Folders within the *\OfficeFX\Repository\PowerPoint* folder define Theme Packs. The Assets contained at this level are usable by all FXThemes within the Theme Pack folder. The Theme Pack folder name does not have associations with other FXTheme files but is stored along with the theme name in PowerPoint when the theme is used by the presentation.

Presenter allows the user to browse within each pack for themes.

Organizationally, the Theme Pack is your mechanism for allowing your themes to share assets. Stylistically, the Theme Pack implies that the FXThemes within it are of the same "family" - be that for retail sale, internal corporate use, a portfolio of offerings, etc. Theme Packs named Pro Basketball, Motorola, or Frog Design are examples of possible retail, corporate, or design firm packs.

Theme level

Individual folders within a Theme Pack folder can each contain an FXTheme. Presenter considers the folder a valid FXTheme if it contains a like-named fxml file (*ThemeName.fxml*). Presenter will display a theme "thumbnail" icon for the FXTheme in the Theme dialog of the FXEditor. To be displayed the thumbnail must be a 64x64 bitmap in .jpg format, with the theme's name (e.g.: ThemeName.jpg), and be stored in the theme folder. Assets contained at the theme level are only usable by the respective theme.

A theme is generally created when you want a specific look and feel that isn't necessarily shared with other themes. For example, a Pro Basketball Theme Pack might have Lakers, Magic, and other teams as themes within it. These would be team specific FXThemes that had no need to share team assets with one another but would share NBA logos, etc. from the Pro Basketball Theme Pack level (see Fig 1.) Similarly, Motorola might have a FlipPhone FXTheme for a new product presentation and Corporate Overview FXTheme for an analyst review that are independent of one another, but share corporate logos and introductions from the Motorola Theme Pack level.

Variation level

Variations reside within the Theme folder as individual FXML files and are theme specific. Variations have the ability to inherit or replace any or all of the attributes of the theme to define subtle or distinct differences in appearance or behavior. Variations are assignable by the user on a slide by slide basis. The FXDesigner is used exclusively to create and modify a theme's Variations and manages their special naming (which is *ThemeName.VariationName.fxml*) so you should never have to manage Variation file naming yourself.

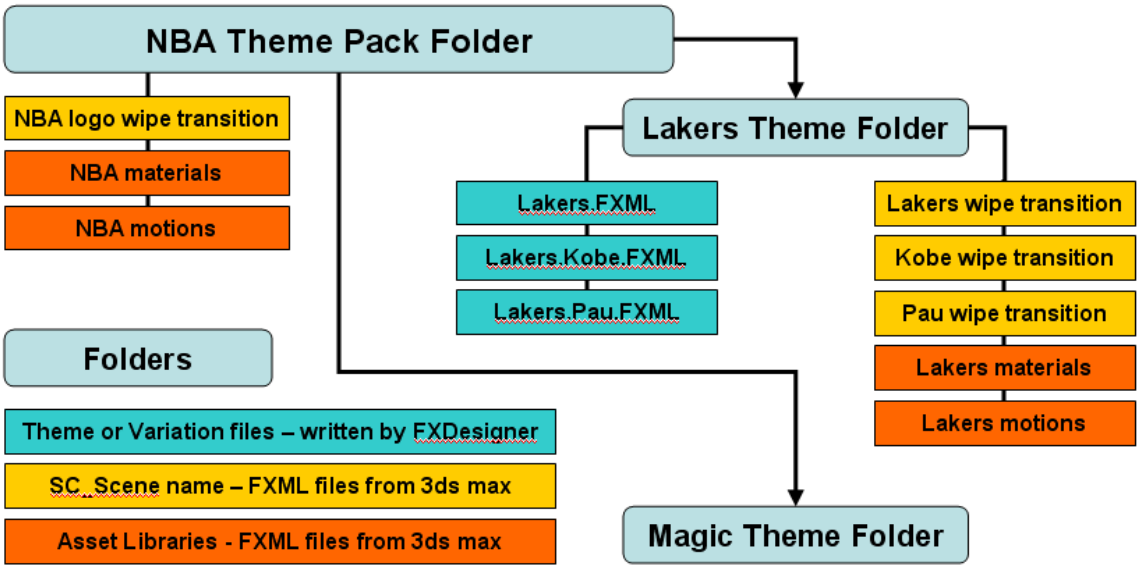


***Fig. 1** – Theme Pack file structure for the Pro Basketball Theme Pack. The Lakers FXTheme is expanded showing the Shaq and Pau Variations as well as Lakers specific scene and asset files.*

Figure 1 shows an expanded view of the Lakers FXTheme within the Pro Basketball Theme Pack. Additional themes would be for other NBA teams (e.g., Magic shown, but not expanded). Assets specific to the theme are stored within the theme folder, while NBA items for use by all teams are stored at the Theme Pack level so all themes can use them. The blue elements of this diagram are created by the FXDesigner while the yellow & orange elements are created as scenes and assets by the FXML Exporter.

FXTheme Concepts

Instant Effects Presenter takes advantage of the tremendous graphics display power that is now available on consumer level desktop and notebook computers. Originally developed for computer games, the latest graphics hardware is capable of drawing multiple complex 2D and 3D scenes simultaneously, at real time rates (30+ frames per second), and with highly realistic shading and lighting effects. FXThemes utilize this display horsepower by layering multiple 3D scenes both behind and in front of the user's PowerPoint content to achieve high impact effects that are integrated with, and driven by, the user's own data.

Layering Foreground and Background Scenes

In building FXThemes there is the ability to include multiple background and foreground scenes with the PowerPoint content. As the names imply, backgrounds will appear behind the PowerPoint content while foregrounds will appear in front. Presenter supports up to four backgrounds and four foregrounds in any FXTheme or Variation.

Foreground and background scenes are created and exported from the 3ds max modeling and animation system. They contain all the elements required to define a 3D scene. These include a camera, lights, geometry, materials, and animation.

Asset Libraries and the Main Content Scene

Another function of FXThemes is to define how the user's PowerPoint content will be displayed in Presenter. Within Presenter that PowerPoint content is actually converted to a 3D representation and placed into yet another 3D scene. We call that the Main Content scene. Conversion of the user's PowerPoint content into 3D format for the Main Content scene must be done at run time as it is not known before then what that content will be.

To determine the appearance and behavior of the user's PowerPoint content, FXThemes draw from Asset Libraries. Those libraries contain lists of material descriptions and visibility animations, as well as sets of transform curves (animations of position, rotation, or scale).

Asset Libraries are also created and exported from 3ds max. They are used at runtime to build a complete scene description in which the user's PowerPoint content becomes the scene geometry. The resulting Main Content scene is then layered for display between the FXThemes' foreground and background scenes. An FXTheme contains instructions for compositing all those elements together in a single Presenter display..

FXTheme Files

The FXTheme folder contains a main <Theme name>.FXML file (where <Theme name> is the same as the theme's folder name) that serves primarily as a pointer to other resource files that make up the theme. These include the theme's Variation and Transition definition files as well as references to other scene files and specific entries from Asset Libraries. Items referenced by a theme file but not contained within it are searched for and loaded at run-time according to a search path hierarchy.

Theme Search Paths

In addition to defining what assets are available to a particular theme, the theme and Theme Pack structure also define the order in which those assets are processed so you only have to be concerned with file name uniqueness within a particular folder. When Presenter is called upon to load an FXTheme, it first gathers a list of all the assets the theme defines and then searches for them in the following order:

	Folder	**Asset Type**	**Usable By**
1	\Repository\PowerPoint\ThemePack\Theme	any asset	Just that theme
2	\Repository\PowerPoint\ThemePack	any asset	Any theme in the Theme Pack
3	*Repository\PowerPoint\Media*	bitmaps & audio files	Any theme
4	*Repository\PowerPoint\Defaults*	FXML and geometry	Any theme

Note: a Media folder may be included at the Theme and Theme Pack level to aid in separating bitmap and audio files from FXML and geometry files. When included, a Media sub-folder is searched before its parent folder is.

This search order allows for convenient separation of scenes, assets, and media files without concern over name collisions. Theme-specific assets do not conflict with any other Theme and take precedence over any other media. Theme Pack assets can be used by any Theme in the Theme Pack and take precedence over system assets. System assets located at the \PowerPoint level can be used by any Theme in the system, but are the last to be checked.

The Theme folder always contains the theme's description, variations, and transitions. The choice of where to locate the assets to which these files refer is up to the theme creator, and would be based on the requirements of that particular theme or Theme Pack. The FXML Exporter provides the option to designate either the Theme or Theme Pack folder on each export as a convenience. The destination folder has no impact on what is written, only to where it is written. Theme assets can be moved to different locations in the search path for organizational and access purposes.

Scene Details

Scenes define full 3D datasets and are created from 3ds max using the FXML Exporter. FXML scene files contain definitions for cameras, lights, helpers, materials, object transforms and animation. Scenes are self-contained and can not share any of their definitions piecemeal (e.g., materials, lights, cameras within a scene can only be used by that scene). They can however share maps and geometry. Scenes reference both bitmaps and geometry by name, which are loaded according to the theme search path, and are shared with other scenes if of like name, allowing any number of scenes to employ the same .x file (3D geometry asset) in different ways. Geometric objects are stored as individual .x files, with full support for instancing. When used in a theme, a scene delivers a fully rendered result of whatever it references.

Multiple scenes may be used within any given Variation of an FXTheme, and the same scene can be referenced multiple times for multiple uses. Scenes are used to define backgrounds, foregrounds, and main content, and may also be part of transitions. This can be powerful, but it does mean that careful naming conventions must be used to assure that proper scene elements are used

Insert Scenes are a special case. They are not part of a particular FXTheme. Instead they are 3D scenes that are created in FXML format and can be added to any slide of an Instant Effects presentation. A typical use of Insert Scenes would be to depict a realistic, animated, and interactive representation of a company's products or logo. The preparation of Insert Scenes is nearly identical to building scenes for use as FXTheme backgrounds and foregrounds. Insert Scenes are a good place to start in learning to create custom 3D content for Presenter. They are discussed in more detail in the section on Creating Scenes for FXThemes. The Instant Effects asset creation tutorial covers the building of Insert Scenes in reference to an example 3ds max source file.

Asset Library Details

Unlike scenes, Asset Libraries contain definitions of scene components such as materials, visibility clips (see below), and animation clips. Asset Libraries are created with the FXML Exporter. In that process they are written to a set of specifically named files:

Scene Attribute	**Asset Library Files**
Materials	*Material.fxml* & *MaterialClip.fxml*
Visibility	*FloatClip.fxml*
Object Animation (Position, Rotation, Scale)	*Transform.fxml* & *TransformClip.fxml*

Assets from within the above files are selected for use in a theme from within FXDesigner. Each theme search path level may contain each of the above asset libraries. The assets available to the theme are a combination of what exists at each level for each library type. For example, if a Material library file (*Material.fxml)* is in both the *ThemePack* and a *Theme* folder and one contains materials 1-10 and the other contains 11-20, then materials 1-20 are available to the theme

Clips – for Triggering Events and Defining Behaviors

Clips represent a critical element of FXThemes. This may be a new concept for traditional film and video oriented 3D artists. Clips allow the definition of "event based" animation, and make elements of a theme interactive and nonlinear for the presenter. Clips are animations that are "triggered" by discretionary user actions – usually mouse or keyboard clicks. The most common use of clips will be to create the custom effects and animations that occur during Presenter transitions. Clips are akin to events or behaviors in other real-time systems such as games.

Clips stored within Libraries define discrete animation sequences for controlling object visibility, material properties, or object animation (position, rotation, scale). Individual Clips can be assigned to options within the FXDesigner to define behaviors for content entry and exit animation as well as selection / de-selection. Clips can also define animation that occurs during slide transitions. Here are a few examples of situations where Clips would be used:

- A material that animates from dull to bright: This clip asset would be used as a selection material. When a text bullet point in a presentation is selected by the user, this clip would cause its face material to animate from dull to bright, resulting in its standing out from the other, unselected bullet points.
- An animation that spins an object forward: Similarly, this motion asset would be used for content selection. When a text bullet, or a picture or a drawing object was selected by a presenter using Instant Effects, it would spin forward to a position in front of the others.
- A visibility animation that causes an object to Fade Off: This clip asset (and its opposite Fade On clip) might be used to drive a subtle content transition between slides in an FXTheme. At the conclusion of a slide, the fade Off clip would cause its PPT content to fade away. Then the reverse clip would cause the PPT content of the next slide to Fade On. In this case the trigger event is the presenter calling for the next slide.
- A logo animation that wipes quickly from one side of the screen to the other: This clip based scene would be used to create a logo wipe transition.

Any effect that is triggered or initiated by the presenter is linked to an Asset that is defined in FXML as a Clip. How animation is flagged as a Clip within 3ds max is described in the section entitled FXML Exporter for 3ds max.

Cycling Behaviors

Not every form of animation in Presenter is triggered. When you want animation to repeat continuously you give it a cycling behavior. In 3ds max this is accomplished by assigning an out-of-range controller to it. A cycling animation can be on materials, lights or visibility, or can cause presentation or Theme elements to move. Some examples:

- A light traveling in a continuous circle while varying the color to shed varied illumination
- A foreground scene shifting its visibility from on to off every 10 seconds to flash a watermark
- A rotating reflection map within a material to give the illusion of environmental motion.
- A background scene with constantly animating objects along with a camera that slowly changes its position to give a continuously new view.

All the above are examples of cycling behaviors that will repeat continuously as slides are changed without any specific input from the presenter. This is in contrast to triggered behaviors that are initiated by some user action (and driven by Clips). How cycling animation is created within 3ds max is described in the section entitled FXML Exporter for 3ds max.

Unified Timeline

Presenter maintains a unified timeline on which all scene events on all slides are placed with time=0 at the start of a presentation. Objects with out-of-range controllers all start their sequence at the same time and run constantly, at the same rate, throughout the entire presentation. This consistent time management allows animating objects and properties to stay in sync with one another as slides are traversed so no “popping” or motion resetting occurs as a new slide is brought up. For example:

- Animated lighting within a Main Content scene that is used in more than one Variation will maintain smooth lighting between the slides that use those Variations.
  Note: Main Content scenes control the view perspective and lighting of PowerPoint content. They are described in more detail further along.
- Mesh objects that have the same animation cycles will stay in sync with one another from slide to slide – even if they are part of different scenes and in different variations.

Objects which have non-cycling animation (do not have out-of-range controllers) will play their animation on the unified timeline just like cycling animation. They will start playing at the beginning of an Instant Effects presentation but then remain static. Even going back to Slide1 later in the presentation will not “restart” that non-cycling animation. As a result this form of animation, which is neither cycling nor clip-based, is seldom used in FXThemes.

Transition Animation

Transition animation is driven by Clips. The transition starts when the event that triggers it occurs. That is typically a mouse click, Enter, or Page Down keyboard command from the presenter. It can also be a PowerPoint timer event if the presentation is set up for automatic slide advance. The duration of the transition is the length of the clip. When the transition uses multiple Clips (e.g., a scene contains multiple objects containing Clips or multiple scenes contain Clips) then the transition will last the length of the longest clip. Details on creating scenes with Clips are provided further on.

PowerPoint Content Animation

PowerPoint content animation in Presenter is also driven by Clips. Content animation can be the entrance or exit of PowerPoint text, drawing shapes, or images. FXThemes also typically provide emphasis animations that allow the presenter to highlight a particular element of a slide's content. The actions associated with that highlighting are termed selection and de-selection. The animations that drive all these forms of PowerPoint content animation are based on Clips. The start of the event is usually a user action, and the duration is the length of clip.

The FXML Export Utility for 3ds max

The techniques we outline in these documents for creating FXML scenes and assets are described in the context of 3ds max authoring workflow. All of those methods employ the Instant Effects FXML Exporter utility to generate the scene and asset files that are the building blocks of FXThemes. As a first step to describing the FXTheme creation process we will introduce the 3ds max FXML Exporter utility.

The FXML Exporters supplied are Utility plug-ins for 3ds max. Supported 3ds max versions are 7, 8, 9, 2008, 2009, and 2010. To support this range of 3ds max releases, three different FXML Exporter Utility programs are supplied with the Presenter Authoring Toolkit. One works with 3ds max versions 7 and 8. One works with 3ds max 9 and 2008. The third runs with 3ds max 2009 and the fourth with max 2010. All are located in the "Executables" folder included with this Authoring Toolkit. To install and run the correct FXML Exporter Utility:

1. For 3ds max 7 or 8 copy the executable file called FXMLUtil6.DLU to a folder in your Plug-Ins search path (e.g., "plugins"). For 3ds max 9 or 2008, do the same with the file called FXMLUtil9.dlu. If you are running Vista and/or 3ds max 2009 use FXMLUtil2009.dlu and for 3ds max 2010 use FXMLUtil2010.dlu.
2. Launch 3ds max and open the Utility Panel
3. Click on More to choose the FXML Exporter. Result: you should now have the plug-in shown in Figure 2

Tip: for quicker access to the exporter session after session, choose Configure Button Sets (upper right of the Utility Panel) and drag the FXML Exporter name from the list onto a button slot of your choosing on the right.

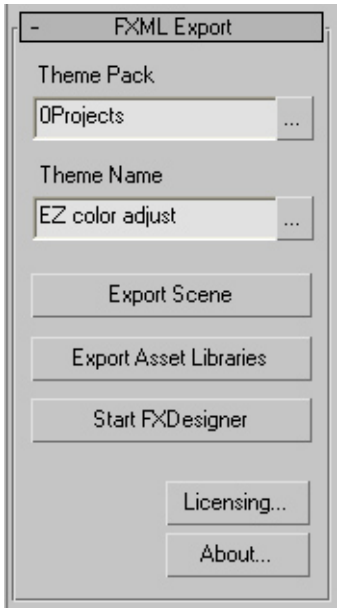


Fig. 2. The FXML Export Utility within the 3ds max Utility Panel.

The fields for Theme Pack and Theme Name define the folder destination for the Utility's actions. The names entered in these fields reference folders within \OfficeFX\Repository\PowerPoint. The Exporter will create these folders when first exporting if they do not already exist.

The *Export Scene* and *Export Asset Libraries* buttons launch different dialog The *Start FXDesigner* button launches the FXTheme Designer with your current theme as its default. The FXDesigner is a stand alone application located in your \Instant Effects\OfficeFX folder that can also be used outside of 3ds max. See the section below on the FXDesigner application.

Insert Scenes are also created with the Export Scene dialog but Insert Scene export does not use Theme Pack and Theme Name folder names. Exporting Insert Scenes creates new Insert Scene folders in \OfficeFX\Repository\Insert.

The FXML Export Utility will work in evaluation mode for 30 days. After that period it must be purchased and licensed for use.

FXML Export Utility – Export Scene

The Export Scene option exports all visible (unhidden) geometric objects within the 3ds max file. It also exports all other objects contributing to the rendered scene – including lights, cameras, and helper objects, *regardless of whether these objects are hidden or frozen.* This ensures all objects contributing to what the viewport shows are captured upon export, and objects commonly hidden (e.g., lights, cameras, parent nodes, etc.) are not accidentally missed. The scene composition, object definitions, transforms, and materials are stored in the resulting FXML scene.

Bit maps used to define "texture" components of materials used in the scene are also gathered and exported.

Tip: Objects with their Renderable property turned Off are always skipped by the Exporter. This is a convenient method to have certain objects unhidden but not exported.

Note: Hidden objects are still recorded in the resulting FXML scene in case they are required for additional references. While the objects will never appear in the scene, they do bloat the file and make it more difficult to edit in the future. When your theme is finished, it is recommended you delete any unused objects before exporting so your FXML scenes have minimum weight and maximum clarity.

Scenes are exported to FXML scene files and contain everything but mesh descriptions, which are exported as individual .x files per object. For scene geometry to be viewed and rendered within Presenter, scenes must contain at least one camera and one light source.

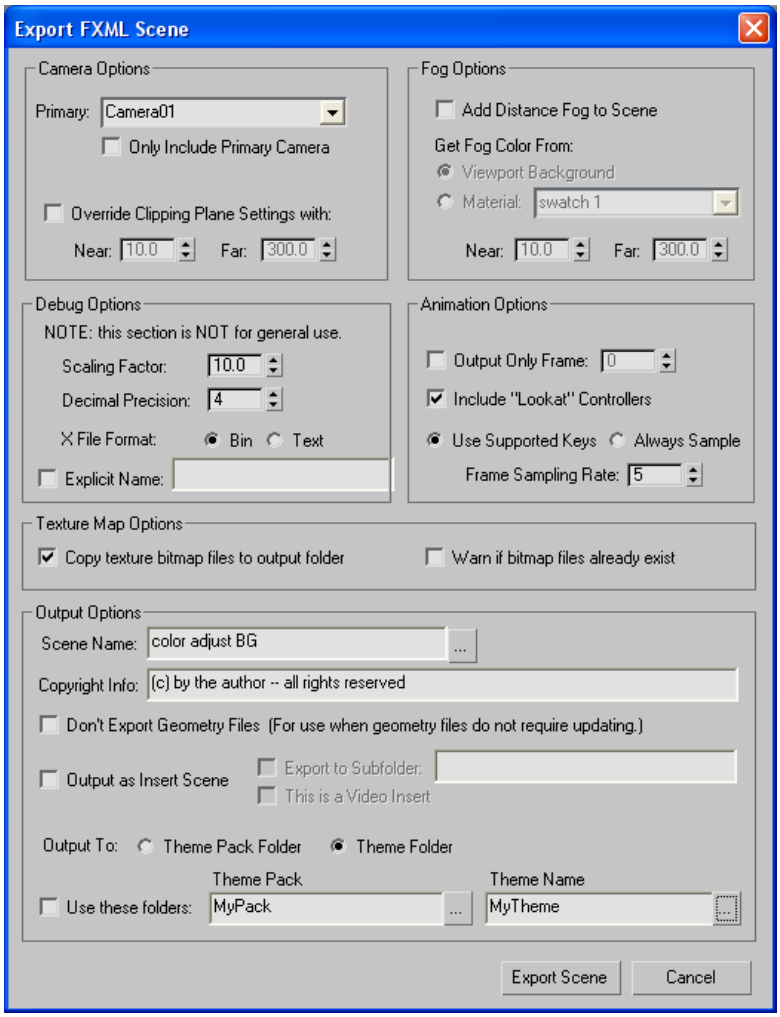


Fig. 3 *The Export Scene Dialog of the FXML Exporter, with default settings and showing output of a new scene file called MyScene going into the Theme Folder Camera Options*

Camera Options

Each scene must have a camera so Presenter knows how to view its contents. By default, every camera within the file is exported but only one camera, the one set to be *Primary*, is set to be active.

Only Include Primary Camera

Exports only the camera defined as “Primary” to the scene file.

This option is **required** to ensure the proper camera is employed when exporting to scenes to be used as backgrounds and foregrounds, as there is currently no easy way to switch cameras after export. Conversely, when exporting Insert Scenes, this option is often **not** desirable because being able to choose multiple vantage points is sometimes an option you would like to afford the user.

Clipping Planes

Clipping planes are critical for real-time (hardware rendered) systems such as Presenter because they limit the dynamic range and increase the accuracy of what is viewed. A clipping plane must be established for every scene export either by enabling Clip Manually in the camera object's Modifier Panel and properly setting its Near and Far Clip (recommended), or by overriding it with the *Override Clipping Plane Settings* option and using the Near and Far values in the Exporter. You will receive a warning on export if the Override is Off and the camera does not have Clip Manually enabled.

Fog Options

When enabled, the Fog option gives the scene a linear fog effect within Presenter. The range settings within the Exporter are similar to the camera's near/far settings for controlling linear fog in 3ds max, but the Exporter is *not* referring to file's camera for *any* fog range information – these are all set within the Exporter. The "Viewport Background" color refers to the 3ds max file's Environment Background Color. The "Get From" option takes the diffuse color from a material assigned to a **renderable** object in the scene.

Debug Options

These section controls are not for most exports, but do have periodic uses.

Scaling Factor

This assigns a general scaling factor to the entire exported scene. The scene's size is divided by the number you have entered as a Scaling Factor. For example, with a default scaling of 10.0, the default camera to target distance of 175 units is recorded in the FXML file as 17.5 units. It's not advisable to change this value unless you are having problems with files of very different scales coordinating correctly in Presenter.

Decimal Precision

This controls the accuracy of the resulting FXML file data. Larger accuracy increases the size of the file but can lead to better accuracy. Increase this value if you need finer evaluation in the runtime for your scene.

X File Format

Exporting to Bin vs. Text (binary vs. ASCII) versions of .x format is used primarily for debugging and not for general use.

Explicit Name

The Explicit Name check box and its accompanying text box are features for advanced FXTheme development. They are used for creating stand-in scenes that are *referenced by* other scenes or that *replace* other scenes in FXThemes. When checked, the scene's external name (on disk) does not get its usual leading "SC__", and the scene's internal (in file) scene name is the one specified here by the user.

Animation Options

This section controls how keys are written to the scene file.

Output Only Frame

Exports a scene with all scene elements frozen at a specified frame.

Include Look At Controllers

Exports relationships for Look At controllers that cause objects to automatically orient to another object within the scene. This option is On by default since target cameras and spotlights both use Look At controllers.

Use Supported Keys vs. Always Sample

Controls whether explicit keys are written for animation controllers that have matches within Presenter or whether those supported controllers are sampled. The *Frame Sampling Rate* is the frequency at which keyframes are written to the file, regardless of where the actual keys were placed within the 3ds max file. Unsupported animation controllers are always sampled. In practice, *Always Sample* should only be used if you are troubleshooting a scene where the resulting Presenter animation isn't matching what is in 3ds max.

Texture Map Options

These controls specify whether the bit maps used in the 3ds max are gathered and exported along with all other elements of the scene. The user can optionally choose to be warned so that bit maps that have been customized in the scene folder do not get over-written.

Output Options

This section controls how and where the scene file is written.

Scene Name

Specifies the name of the FXML scene file. Exporting to a Theme or Theme Pack produces a file with a "SC_" prefix while Insert Scenes are exported without a prefix. If you use the file browser to select a "SC_" prefixed name to update, the browser will strip the "SC_" from the name for use in the Exporter FXML Scene dialog.

Copyright Info

An optional field to include Author information in the header of the exported FXML scene.

Don't Export Any Geometry Files

When enabled, the FXML scene is exported but no .x files are written. This option is useful for making rapid design iterations that do not require updates to previously exported geometry (topology, mapping coordinates or pivot point changes). Any changes to object transforms, animation, materials, lights, cameras, and fog will export while not exporting geometry.

Output as Insert Scene

Checking the "Output as Insert Scene" option will export the file as an Insert Scene. The specified scene name will be used. If it does not already exist, a folder with that scene name will be added to the OfficeFX\Repository\Insert folder. Results of the export will be placed in that folder.

Checking the "Export to Subfolder" option will allow you to create a grouping of Inserts Scenes. These can be the logos or products of a particular client or the Inserts Scenes assets associated for a particular product line within your company. If you produce Inserts Scenes for a variety of customers, it is highly recommended that you use Inserts Scene subfolders.

Checking in addition the "This is a Video Insert" option will export the scene as a Video Insert Scene and place the result in OfficeFX\Repository\Video Insert. The "Subfolder" option works the same for Video Insert Scenes.

Output To: Theme Pack Folder, Theme Folder, Insert Folder

By default, the FXML Export utility writes its output scene and .x files to either the Theme Pack or Theme Name folder specified on the FXML Export panel in 3ds max. This option allows you to specify which of those folders are used. Recall that files output to the Theme Pack folder will be available for use in any FXTheme within that pack whereas files output to the Theme Name folder will only be available for use in that FXTheme. With this control you can reduce file duplication (hence minimize the size of your finished Theme Pack).

Use theme folders:

If the “Use these folder” option is checked then the Theme Pack and / or Theme Name options that are listed right there in the Export FXML Scene dialog are used instead of those folder names from the FXML Export panel in 3ds max. Note that the Output To option just above also controls whether the export goes to these locally specified Theme Pack or Theme Name folders.

FXML Export Utility – Export Asset Libraries

The Export Asset Libraries panel of the FXML Export Utility creates asset library files for materials, animation (transforms), and visibility, As with Export Scene, the resulting asset libraries can be exported to either the Theme Pack or Theme folder.

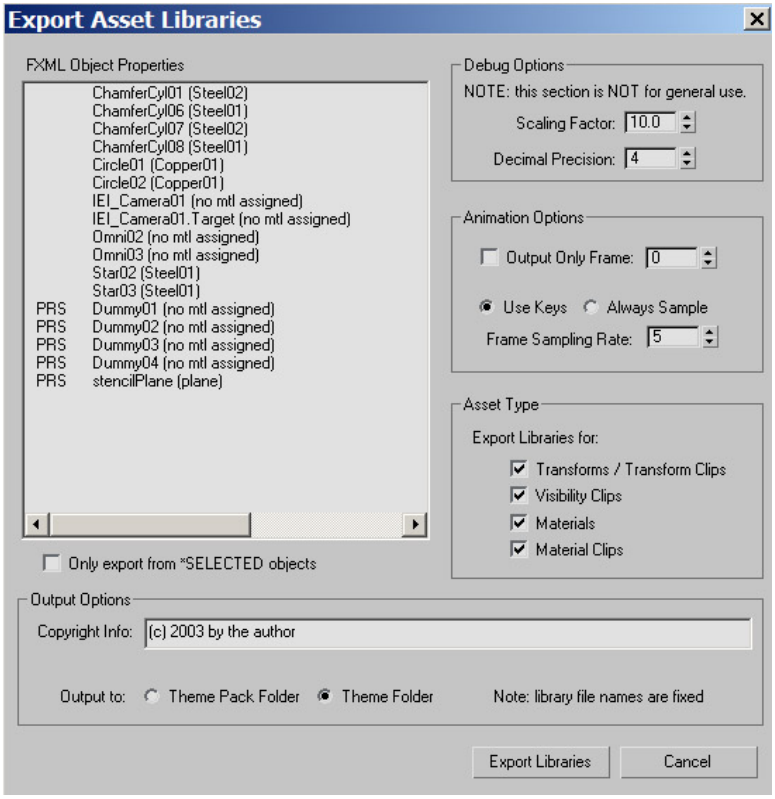


Fig. 4 *Asset export panel showing settings for exporting Transform and Transform Clips, Visibility Clips, Materials, and Material Clips to the specified Theme folder. FXML Object Properties*

FXML Object Properties

This section lists the objects within the 3ds max file. The left column lists Clip assignments, while the object’s material is listed in parentheses after the object’s name. Items with an asterisk (*) indicate objects that are currently selected.

Debug Options

Scaling Factor

Same as within Export Scene

Decimal Precision

Same as within Export Scene

Animation Options

Output Only Frame

Same as within Export Scene

Use Keys

Same as within Export Scene

Output Options

Copyright Info:

Same as within Export Scene

Output to:

Same as within Export Scene

Asset Type

Export Libraries for:

Determines which asset library files are written upon export. Each item checked will output the respective library, and you can export as few or as many at one time as is convenient for you.

***Note**: Asset libraries are exported to specific file names (listed below) which cannot be changed by the user. As a result, each export overwrites previous versions of those file names – and does so without notification. Exporting always overwrites files and does not append or update existing ones.*

***Tip**: It's best to keep a single 3ds max file for managing the clip and material assets as the master source for any given theme or Theme Pack. You can then add and update the animations and materials in the 3ds max file and easily export updated libraries.*

Scene and Asset Naming Conventions of the Exporter

The files created by the FXML Exporter have the specific conventions noted in the below table. FXML file names within quotes are controlled by the person generating the FXTheme based names used within 3ds max or the Exporter. File names without quotes are created automatically by the Exporter or FXDesigner during the process of FXTheme construction. They are reserved file names that are the same for any FXTheme.

FXML File Name	Name Defined By	Contains
"ThemeName".fxml	FXDesigner. A like-named Theme folder is created at same time.	Theme Definition and pointers to other theme files and assets
SC_"ExportedSceneName".fxml	FXML Exporter. When exporting Exporter adds "SC_" prefix	Standard Scene
"ExportedSceneName".fxml	FXML Exporter. When exporting a new folder and scene is created in OfficeFX / Repository / Insert	Scene for use as Insert
"MaxObjectName".x	FXML Exporter. .x file gets name of the 3ds max mesh object	One mesh object and mapping coordinates
Material.fxml (Materials within it)	FXML Exporter using the name of material, assigned to each non-Clip mesh object in the file.	All static materials (materials without Clips) are exported
MaterialClip.fxml (Clips within it)	FXML Exporter using the name of the material assigned to each mesh with Clip=M property in the file.	All animated materials with Clip user defined property
FloatClip.fxml (Clips within it)	FXML Exporter using the name of the material assigned to each mesh with Clip=V property in the file	All visibility animation Clips from objects with Clip=V user defined property
TransformClip.fxml (Clips within it)	FXML Exporter using the name of mesh objects with user defined Clip=PRS properties	Transform animations (position, rotation, scale) that can be triggered by user input.
Transform.fxml (Transforms within it)	FXML Exporter using the name of mesh objects with non-Clip animation	Transform animations that play automatically without user triggering
Transitions.fxml (Transitions within it)	FXDesigner when Custom Transitions are defined	Definition of FXTheme's Custom Transitions

FXML Exporter – Supported 3ds max file properties

This section explains what properties within 3ds max are supported on export to FXML.

Geometry

When exporting scenes, objects are exported as individual .x files containing the mesh seen in the 3ds max viewport. Note: what is exported is the current end result of the modifier stack. This allows for deformations and even character skinning.. The scene's actual FXML file simply makes reference to these external .x files, allowing multiple scenes to use the same .x files with different transforms, animations, and materials. Instanced objects create only one .x file but are recorded separately in the scene file so they behave just as they did in 3ds max.

The tessellation, smoothing, and mapping for all objects are translated as they appear in the 3ds max viewport at the time of export. Non-contiguous meshes (3ds max "elements") are exported within the one .x file for the object, along with the mesh's mapping coordinates and smoothing group assignments.

Objects in Presenter can have more than one material per object. The FXML Exporter Utility properly exports Multi-Subobject Materials. Vertex color information is not exported.

Shape objects are only exported if they are currently displaying a mesh in the viewport. Meshes that are hidden or whose rendering properties are turned "Off" are not exported. Helper Objects are exported as non-rendering nodes. Hidden faces within EditableMesh objects are always exported if the object is visible.

Note: The base shader within Presenter is Gouraud based, calculating its light on a per-face level similar to the 3ds max viewport renderer. Objects reflecting light poorly in the 3ds max viewport are a good indication they need adjustment and/or finer tessellation for proper rendering by the Presenter base shader. An advanced option for avoiding this, would be to use a DX9 shader that employs a per-pixel rendering method (e.g., Phong or Blinn).

Tip: Be careful with generic mesh names – especially when you build up scenes using multiple primitives like cylinders or planes. 3ds max assigns default names like Plane_01 or Cyl_01 to those meshes. It's recommended that your change those generic names to something specific to the scene (like Floor, or flagpole). Doing so will avoid the problem of subsequent scene exports overwriting .x files with the same names.

Materials

The Exporter supports the Standard materials of 3ds max. It associates certain values you set within the Material Editor's Standard material with those of the Base level shader of Presenter. In most cases, this is a mapping of values and not an exact rendering match to what the max renderer or viewport displays. All objects must have a material assigned to them to export.

Note: Specialized materials such as the Architectural materials supported by 3ds max must be converted to Standard materials to export correctly.

Base Shader Associations

By default, all material definitions assigned to objects are mapped to the base Presenter shader and included within the scene file on export. This base shader uses the DX7 fixed function pipeline and Gouraud rendering for maximum hardware compatibility. The following traits found within the Standard material are matched to close equivalents within the Presenter Base shader on export:

- Ambient, Diffuse, and Specular colors
- Specular Highlight Level, Glossiness, and Softness
- Opacity percentage and Filter, Subtractive, Additive modes
- Self Illumination (a.k.a. emissive) percentage and Two Sided option
- Diffuse, Reflection and Opacity map assignments and percentages

- Bitmap Map and controls for offset, tiling and rotation
- Tile On / Off (for producing “decal” type texture maps)

No other Map slots, Map types, or Material types are supported at this time due to the limit of Direct3D’s fixed function pipeline. Additional parameters (e.g., Bump or Metal) can be added as DX 9 shaders.

Other Shader Associations

Alternative shaders can be assigned to objects by giving them the User defined Object Property *ShaderIncludeFile=shadername.xml* (or *SIF=shadername.xml* for short) where the .xml is an association file that maps the supported Standard material values to Tweakable parameters within a corresponding shader.fx file. Details on creating custom shaders are provided in the Applying Shaders segment of the Advanced Theme Creation Topic tutorials at the end of this document.

Bitmap Map Support

Many items of the Bitmap Map’s Coordinate Rollout are respected by the Exporter:

- Offset, Tiling and Rotational values are exported, but the origin is at the upper left corner in OfficeFX rather then the center as it is in the 3ds max Bitmap map. Because of this, the results from the Bitmap Map’s tiling and rotation will not match what you see in Presenter, although the results for seamlessly tiling textures will be very close.

Tip: If you have specific map alignment needs, you will achieve more accurate results with a modifier than from controls within the Bitmap Parameters section of the Material Editor. The Map’s parameters are of most use when the bitmap seamlessly tiles and/or alignment of a particular point is not required. Note that the UVW XForm modifier works well for adjusting mapping that is already baked into the mesh.

- The On / Off check box options for the U and V Tile settings are exported . The default value for these options in 3ds max is On. Turning them Off causes the bitmap’s border RGB and Alpha (opacity) values to be used everywhere else on the mesh other than where the one copy of the texture is placed. When used as both opacity and diffuse texture maps this feature is particularly useful for generating decals and labels on mesh elements.
- The Mirror option is not exported, as there is no concept of a “book matched” map in Presenter at this time.
- The Texture Map Type option is not exported, as Diffuse and Opacity maps are always “Explicit Map Channel” and Reflections maps are always “Spherical Environment”.
- Blur options are not exported. Motion blurring is not a feature supported by Presenter.

Other Rollouts of the Bitmap Properties dialog in 3ds max are not supported by the Exporter.

Bitmap Format Support

The bitmaps used by material maps (diffuse, reflection, opacity) are included in the scene file as named references. Please note the following regarding bitmap files:

- All bitmaps must be within the Theme’s search path. When launching Presenter, bitmaps not found in the search path are skipped and will register an error message in the Diagnostics.
- Bitmaps used within Presenter are translated to the closest dimension that is a power-of-two (e.g., 256x256, 512x256, 1024x64, 512x1, etc.) for optimum speed and memory usage. Advanced users generating highly complex themes will achieve better performance and use less memory by using bitmaps whose actual dimensions are powers-of-two.
- Transparency information (for Opacity maps) is only respected as the alpha channel of a 32-bit format file (e.g., TGA or PNG). The RGB color information within the bitmap is always ignored when calculating Opacity, although the bitmap could be used in another capacity (e.g., Diffuse) where color is used.

Tip: Where the compressed (run length encoded) version of TGA files is used, that format is recognized by Presenter. By default however, TGA files are not compressed and can be quite large. Default PNG files are compressed. With no sacrifice in image quality they are significantly smaller.

- Animated file textures are supported, but are restricted to specific file formats. Presenter supports AVI, WMV, MPEG-1 and MPEG-2. It does not support QuickTime files (MOV). 3ds max supports both AVI and MOV formats, but does not support WMV or either of the MPEG formats. When an AVI file is mapped to the defuse component of a material that is used in a scene, that AVI file will playback in the scene when it used in Presenter. The FXML Exporter will include the AVI file with the assets it exports.

Note 1: Presenter respects video compression / de-compression (codec) utilities that are used in the playback of animated file textures. If the Insert Scene or FXTheme that uses such video files is to be transported to any other computer, that same codec must exist on the new machine.

Note 2: This capability of authoring scenes where a specific video plays back, is different from the Video Insert Scene capability of Presenter. Video Insert Scenes insert scenes are more flexible. They allow Presenter end users to select any video that is in one of the supported formats for playback at runtime.

- File formats not respected by Direct3D (e.g., TIFF) are not supported by Presenter for use in themes. Files such as TIFF loaded into PowerPoint as presentation content are supported because a generic bitmap is fed to Presenter rather then the original TIFF file.

Lights

Up to eight light objects are supported per scene. The number of lights in one scene does not impact the number of lights other scenes may have that are being used to create the final display. The following aspects of Omni, Spotlight, and Directional light objects (free and target) are exported:

Color, Multiplier, Hotspot & Falloff

The color and multiplier are exported for all light objects. Hotspot and falloff values are exported for spotlights but not directional lights, as the latter are always considered infinite.

Near Attenuation

If enabled, Near Attenuation values for omni and spotlights are respected in Presenter for a linear falloff effect, but Far values are not supported. Thus, you can state where attenuation will begin, but it will never “end”. This can provide a good tool for more sophisticated scene lighting.

Unsupported Parameters

Only positive multiplier numbers and circular spotlights are supported. Projector maps and shadow parameters are not exported. Ambient Light is also not currently exported.

Clips for Lights

Lights can be only be assigned Clips for position and rotation. To obtain the effect of a dimming or color changing light, you can use a plane with a visibility or material clip in the front of the scene to act as a camera “filter”.

Cameras

Free and Target cameras are both supported, along with their FOV and clipping plane parameters. Cameras can be only be assigned Clips for position and rotation. Animating FOV or clipping planes is not currently supported. The orthographic option is not supported.

Tip: Enabling and setting clipping planes gives you a WYSIWIG view of how your camera will view your scene in Presenter, and is recommended for all exports.

Animation Controllers

The natively supported controllers within Presenter are Bezier Float and Linear Float. All other controllers will be interpolated according to the frame sampling rate set in the Exporter. Controllers that depend upon other objects (e.g., Path controller) are always sampled, with the relationship no longer tracked after export.

Tip: To maintain key frames and avoid sampling you are advised to use Position XYZ, Scale XYZ, and Euler XYZ as your default controllers so you can easily assign Bezier or Linear Float controllers to the resulting components.

Note: The Bezier controller isn't an exact match, and you may experience slight inconsistencies (such as overshoot with widely spaced keys) within Presenter. When this happens, it's both informative, and a possible fix, to use the "Always Sample" option to study how the keyframes are being interpreted.

Out-of-range Controllers

The out-of-range controllers for Linear, Loop, PingPong, RepeatRelative, and Cycle are all supported.

Tip: You will be assigning out-of-range controllers for anything that needs to animate constantly within a scene. In some cases it's useful to employ path constraints to control a cycling animation of some scene element such as a light. The resulting animation gets represented as a "parameterized" value. In the case of path animation, the value is percentage of progress along the specified path. Setting out-of-range controllers on such parameterized valued in 3ds max does not cycle their animation in Presenter. They will travel the path for one cycle then stop. In order to get path animated objects to cycle properly in Presenter you must first use the "Collapse Controller" tool to convert their animation to actual position data, then set your out-of-range controller type on that positional information.

Scene Export Check List

The following should be considered each time for successful scene exports:

- Light(s) – at least one that's adequately illuminating the scene
- Camera(s) – at least one that's looking at the scene
- Camera(s) – are the clipping planes active and properly set for the scene?
- Objects – do all objects have material assignments?
- Objects with alternative shaders assignments – does each have mapping coordinates?
- Are the Theme and Theme Pack names correct?

Creating Scenes for FXThemes

Scenes can be referenced in the FXDesigner for use as Foreground, Background, or Main Content scenes. The Main Content scene defines the look of the PowerPoint content that is generated at runtime. Presenter can then (at runtime) add up to four additional Foreground and Background scenes that are composited by Presenter in relation to the PowerPoint content. At its most basic, this can be thought of as traditional layer compositing as shown in Figure 5.

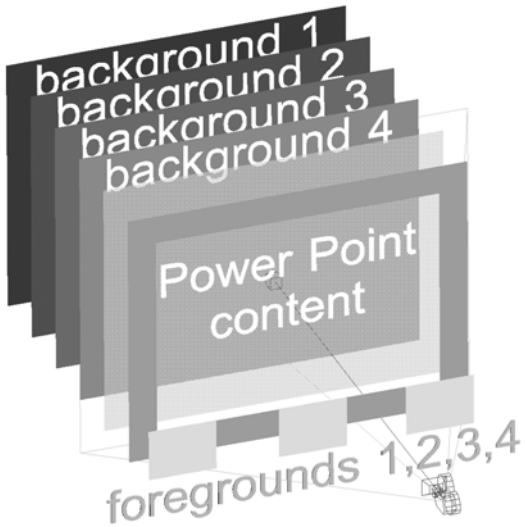


Fig. 5 *Presenter background and foreground scenes viewed as a traditional composite.*

Note: this documentation uses the term "scene" to refer to what is exported from 3ds max and is later used by an FXTheme, and "3ds max file" to refer to content within 3ds max. For example, "load the 3ds max file and export a scene".

Each scene is a self contained composition that is composited back to front by Presenter from Background to Main Content to Foreground. For example, you start with your Main Content scene – perhaps with slowly moving lights to subtly shift the highlights on the PowerPoint content that they will illuminate. Behind this you may place a background scene having a distinctive floor plane and fog to indicate depth, and in front you may include a foreground scene containing a corporate or product logo.

While it's possible for scenes to replicate the composite shown in Figure 5, it would mean that each scene contained a single flat image, and that higher number (later composited) background scenes would need to be transparent or they would obscure those behind. In most cases Presenter scenes are 3D in nature and take on the behavior shown in Figure 6. Here two Foreground scenes are rendered in front of the Main Content scene which in turn is in front of a 3D background scene.

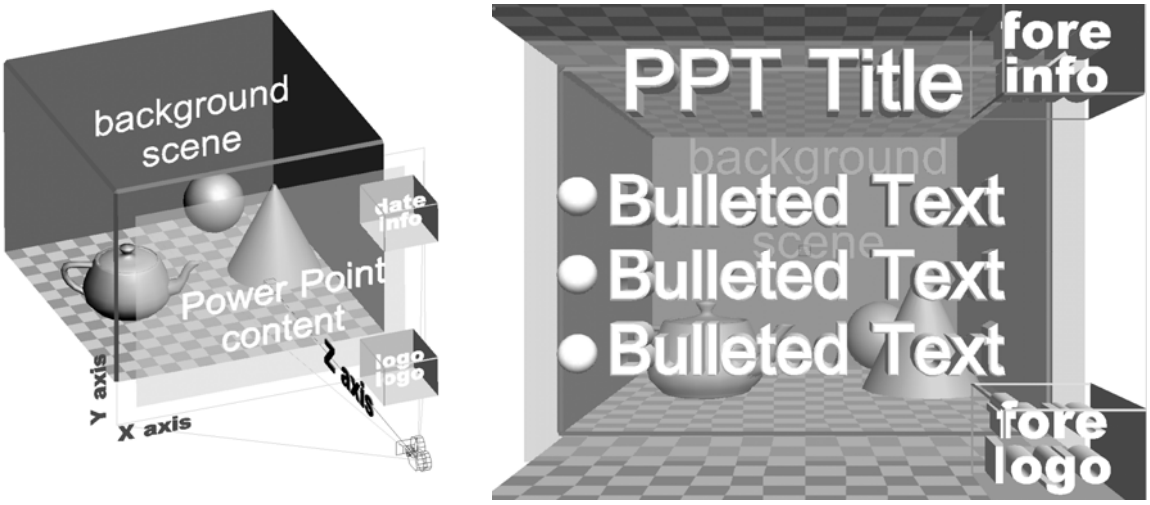


Fig. 6 *Layered 3D scenes in concept (left); and when rendered at runtime (right)*

What Figure 6 doesn't illustrate is that each scene can be very independent as to its scale, orientation, and angle of view. For background and foreground scenes, all that matters is what is framed by that scene's camera. The result can be very much like overlaying pictures taken by various camera locations and lenses, but since these are 3D scenes, the results blend with one another according to depth and opacity and do not have hard borders.

Note: An exception to this generality will occur when advanced users build stereo 3D (S3D) content. In that case best results are achieved by using a consistent viewing space for all scenes that will make up the final FXTheme.

Setting up Scenes

Scenes can be used by an FXTheme as either a Background, Foreground, or for Main Content. The designation of how a scene is used is done within the FXDesigner rather than when exporting from 3ds max. This allows the same scene to be used in different ways by different Variations or Custom Transitions, or even by other FXThemes (if saved to the Theme Pack folder).

Even though the Z depth of the scene is determined by its Background / Foreground ranking in the FXTheme that references it, the placement of the scene (in screen X & Y) is controlled by the scene's camera. For example, if the logo shown in Figure 6 was changed from Foreground to Background, its placement and appearance would not change, but the Main Content scene would now render your PowerPoint content in front of it rather then behind it.

Coordinates within an Presenter presentation are oriented according to the camera with the Camera to Target axis being Z, and the X & Y axes being the left/right and up/down of screen space. In general, what your camera views in the 3D creation application will be what Presenter displays. If, for example, the camera views objects in the upper right corner of its view, then Presenter will locate those objects in the upper right of the presentation.

The scene's physical size does impact the result in two ways. First, all cameras in 3D tools used to create scenes have near and far clipping planes to improve performance and accuracy. Presenter respects these clipping planes. Parts of the scene extending beyond the far clipping are clipped and not rendered in Presenter. Second, greatly differing scene scales can cause unexpected compositing errors between scenes and/or main content during animation. If you see such errors then you will need to reconsider your scene scale(s). Scene scale is coordinated by the 3D creation application used to create the scene. With the 3ds max FXML Exporter, scene scale can also be adjusted during the export process. In general however it is recommended that you build FXThemes with scenes of similar scale.

Note: Advanced FXTheme creation techniques allow scenes to share the same Z space. In these cases objects from one scene can interact with objects from other scenes or even with PowerPoint content. This is accomplished by choosing NOT to Clear Z when compositing these scenes. This choice is offered on the Advanced Properties page for base Themes and Advanced Variations in the FXDesigner.

Types of Scenes

Scenes are created by the FXML Exporter, and once created, can be referenced any number of times by Themes and their Variations and in numerous capacities. That referencing is done in the FXDesigner application.

Background & Foreground Scenes

Scenes referenced as Backgrounds are placed behind the Main Content scene with Background1 being the farthest back. The Background1 scene is treated differently then other backgrounds in that it will include the scene's Environment Background Color as its own background while all other backgrounds will ignore this color. Scenes referenced as Foregrounds are placed in front of the Main Content scene from Foreground1 to Foreground4, with Foreground1 being the farthest forward. Scenes always display their entire contents in the presentation and animate according to the unified timeline.

Where Clip properties are added to a scene, those Clip animations will not execute until they are triggered. The user action that will trigger those Clips is determined when the FXTheme is configured in the FXDesigner. Most often, Clip animations in scenes are triggered during slide transition.

Scenes generated for use as Backgrounds or Foregrounds are exported with the FXML export utility. They are exported to either the user specified Theme Pack folder or Theme folders. When exported to the Theme Pack, they are accessible to all FXThemes in that pack. Scenes exported to a theme folder will only be accessible to that theme.

The Main Content Scene

Creating a Main Content scene is optional. Presenter provides a default Main Content scene that precisely matches a full frame PowerPoint slide. If no Main Content scene is specified by an FXTheme, the default scene will be used automatically. It is recommended that your initial FXThemes be created without specifying a Main Content Scene. Once the basic concepts of theme creation are well understood, more advanced effects can be achieved through use of custom Main Content scenes.

The Main Content scene defines the view (camera) orientation and lens characteristic, along with scene lighting that will be used to display the user's PowerPoint content. It's typical for the Main Content scene to include just the defining camera and lighting. In general, no design geometry should be included in Main Content scenes. The geometry for Main Content scenes is created automatically by Presenter from the user's PowerPoint content at runtime.

The default Main Content scene that Presenter employs creates a perfect match between the position of the user's content on a PowerPoint slide and its position in the XY (screen) plane when that slide is viewed in Presenter. Using custom Main Content scenes can change this "WYSIWYG" relationship. There are generally only two reasons for including Main Content Scenes in FXThemes.

1. The theme requires PowerPoint content to be compressed onto an XY viewing plane that is somewhat smaller, or differently oriented than it is in PowerPoint. As an example, the FXTheme called “The Core” that is provided with Presenter places PowerPoint content onto a viewing plane that slides in and out of frame. That viewing plane provides contrast so the content is legible over the background. It is also smaller than the full view. A Main Content scene with a non-standard camera is used to size the PowerPoint content appropriately to fit on the theme’s viewing plane.

2. The theme requires that the PowerPoint content be lit with something other than default lighting. The Presenter default Main Content scene provides a single static, white light that is located over the left shoulder of the viewer. If animated or colored lighting of PowerPoint content is desired, that non-standard lighting is created in a Main Content scene.

Tip: if you desire the user’s PowerPoint content to be placed amongst geometry, perhaps a theater’s stage arch, it would be best to make that geometry its own scene that could be used as a background or foreground to frame the content. Putting that stage arch mesh in the theme’s Main Content scene is not the best way to achieve this effect.

Unlike foreground and background scenes, the correct location and orientation of the camera in a Main Content scene is **critical** because the content coming from PowerPoint is being delivered to a specific world space location by Presenter. Figure 7 shows the camera setup for the Main Content scene required to match the same result delivered by a PowerPoint slideshow. These are the camera settings of the default Main Content scene that will be used by Presenter if no Main Content scene is specified in an FXTheme.

Note that the FOV and camera-to-origin distance relationship shown is required for the Instant Effects presentation result to match PowerPoint’s. You may change the FOV to affect perspective, but will need to dolly the camera accordingly to maintain the same frustum size at the origin so your PowerPoint content remains consistently sized.

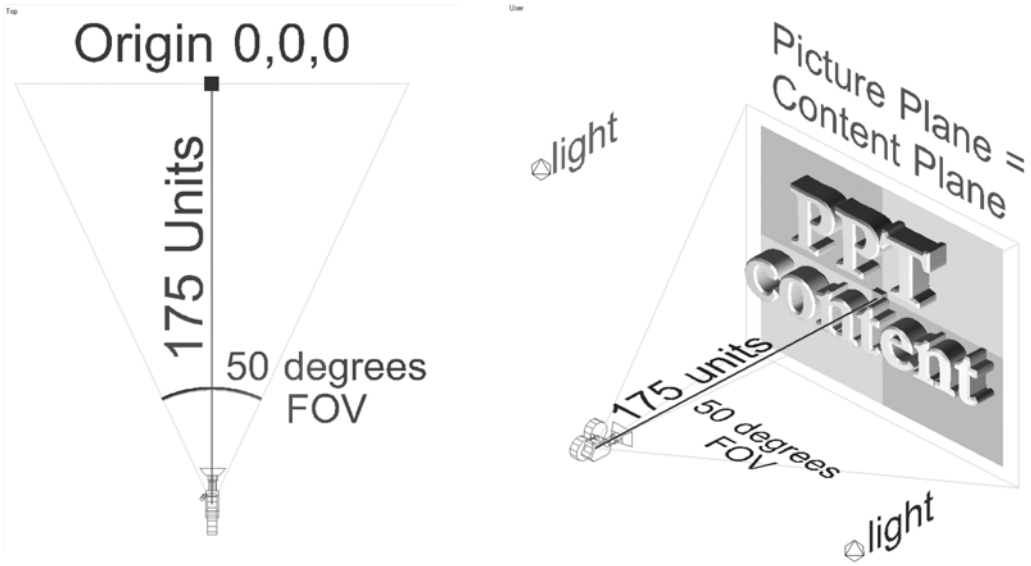


Fig. 7 *Basic Scene Setup for 1:1 relationship with PowerPoint slideshow view.*

The user’s PowerPoint content is centered on the world origin and faced towards screen-Z when Presenter builds it at runtime, regardless of where the Main Content camera may be located or pointing. Moving the scene camera’s location from this state will change the viewing orientation (as in Figure 8) and moving the camera’s target will pan the view of the content – perhaps even out of view. While these results are not often the desired outcome for presentation content, they can be quite useful as advanced techniques in more complex FXThemes.

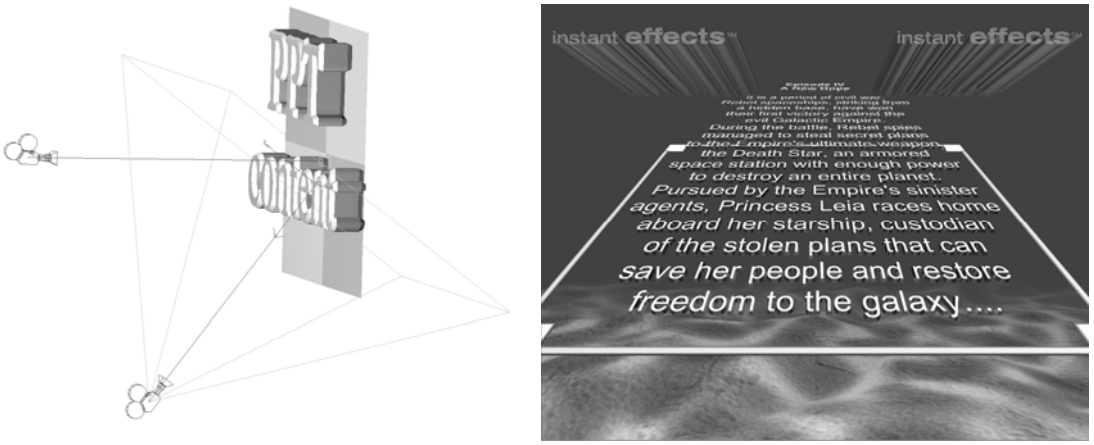


Fig. 8 Moving a Main Content scene's camera, and the resulting view of PPT content.

Figure 8 shows how moving the Main Content scene's camera downward tilts the content coming from PowerPoint while not affecting the perspective of the background Seafloor or foreground Logo, while all three scenes combine as if their content existed in the same 3D scene. The slide content on the right also included a rectangle stretching to the edges of PowerPoint's display area, demonstrating that the Main Content camera can show PowerPoint content that extends beyond PowerPoint's traditional slideshow extents.

The Main Content scene can also contain animation. It's common, for example, to include some animation to lights so there are subtle changes occurring to the presentation content. Animation controllers must be given out-of-range controllers so their results cycle and continue smoothly from slide to slide.

The Main Content scene does not define the materials or resulting geometry of the PowerPoint content. The materials assigned to presentation content are defined in the FXDesigner from Asset Libraries (see material assets). The extrusion profile used to define the presentation content geometry is chosen within the Presenter interface as a slide option.

Scenes and Transitions

Objects having Clip properties will reserve their animation until their scene is used in conjunction with a Custom Transition. Animated objects within the scene without clip properties will animate according to the unified timeline.

For example, a Teapot rotates with an out-of-range controller and animates up and down for frames 1-60. It is assigned a Clip property for Position. When that scene is used as a background, the teapot will rotate continuously but show no position change. When the scene is used in a "Clip" type Custom Transition, the Teapot will move up and down for 2 seconds when the user advances to the next slide. Note that the rotation transform will also continue to play on the unified timeline during the transition.

Insert Scenes in Presenter

Presenter also includes the ability for end users to add 3D scenes to any slide, regardless of theme, via its Insert feature. The compositing order of Insert Scenes can be controlled in Presenter. By default Insert Scenes are displayed in front of your PowerPoint content, but the Presenter user interface provides a "Display behind PowerPoint" option that allows those assets to be composited just behind the Main Content scene. Like any scene, the rendered display from an Insert Scene is the result of its camera, lights, and assigned materials and animation.

To be usable as an Insert Scene, the FXML file must contain header information that is not included when exporting scenes as Foreground/Background, meaning it must be explicitly exported for use as an Insert Scene. Exporting 3ds max files for use as Insert Scenes is an option of the 3ds max FXML Exporter utility. Insert Scenes are exported into folders that take their name from the exported Scene Name. By default, the FXML Exporter utility creates the Insert Scene folders in the OfficeFX\Repository\Insert folder. That is where the Presenter Insert option file browser looks for them first.

Note: Insert Scenes can not be used for any other purpose. If you want your file to be usable as a foreground or background scene, you must export it again to either the Theme Pack or Theme Folder and with the Insert Folder option OFF.

Unlike other scenes in use by Themes, Inserts Scenes actually have no limitations as to where they are located on disk. While Presenter looks for them in its Repository / Insert folder first, it also provides a file browser to get to them wherever they are. One additional requirement that results from this freedom is that all their assets (such as bitmaps used as textures) must be located in their folder with them. More details on the creation of Insert Scenes are provided in the Tutorials section at the end of this document.

Creating Asset Libraries for FXThemes

To restate, within the context of FXTheme building, Assets are data properties that control the appearance and behavior of the user's PowerPoint content when it is displayed in Presenter. This section will provide details on the process of creating and exporting FXTheme Asset Libraries. These Asset Libraries are created in 3ds max and exported to Theme Pack or Theme folders. When Asset Libraries are exported to the Theme Pack folder, they can be used by any Theme in the pack. When they are exported to the Theme folder, they are available only to that Theme.

Asset Libraries are used in the Main Content scenes that are composited in front of the FXTheme's background scenes and in back of the FXTheme's foreground scenes. They determine the look and behavior of the PowerPoint content in the Main Content scene. The individual assets in the Asset Libraries are referenced in the process of creating FXThemes using the Instant Effects FXDesigner. That process will be described below in the section entitled Building FXThemes with FXDesigner.

Assets control the following aspects of PowerPoint content in Presenter:

- Material properties
- Visibility
- Motion - animation of position, rotation and scale

Material and visibility assets can be static values that never change. Alternatively, they can be constantly changing values (cycling behaviors), or they can be the triggered behaviors called Clips. The majority of animation assets that control position, rotation or scale of PowerPoint content are Clips. This is because for the most part those motion assets are used to highlight or make that content appear or disappear based on user input. There are isolated exceptions which we will cover further along, but in general transform animation assets are stored as Clips.

Assets are created in 3ds max and exported using the Instant Effects FXML Exporter. Asset export is initiated from the Export Asset Libraries button on the FXML Exporter Utility in 3ds max (see Fig. 2). As with Export Scene, the user must first enter a destination Theme Pack, and, if desired, a Theme into which the Asset files will be written.

Asset Libraries will typically be written to the Theme Pack. This is because elements of appearance and behavior are likely to be consistent throughout an entire Theme Pack. By writing Asset Libraries to that level in the FXTheme hierarchy, those assets will be available to all Themes in the Theme Pack. In the case where assets are only relevant for a particular Theme, they can optionally be written to the Theme. In that case the target Theme Name must be entered in the FXML Exporter's main menu.

In this document we will treat the creation and export of Materials and Visibility Assets separately from the creation of Motion Assets.

Tip: You will probably find it most convenient to create two separate 3ds max files for building FXTheme Assets. One file would be for your motion assets with the second focused on appearance (material and visibility) properties. The instructions provided here assume this separation of files.

Material and Visibility Assets

Material and Visibility assets are treated as one group for the purposes of these documents. Their creation in 3ds max and their use in FXThemes are similar.

Material Assets

Presenter is a real time rendering system. It does not support all the material properties that are available in 3ds max. The material properties which are recognized by Presenter have been covered previously. The first step in creating Material Assets is to build a set of materials which comply with those constraints. These materials will be used to control the appearance of the following aspects of PowerPoint content.

- Face, side, and bevel materials for both title text and bulleted text
- Picture and drawing shape borders
- Drawing shape text and fill color
- Select and Deselect emphasis colors for all of the above

Use the 3ds max Material Editor to visually tune your materials – see Fig 9.

Tip: You will want to create families of materials that provide contrast over both light and dark backgrounds. Also, make sure to name your materials so they can be later identified by that name when they are chosen in the FXDesigner. FXDesigner alphabetizes name lists so material names like blueLight, blueMedium, blueDark will group together For animated materials (material Clips), the name should describe the dynamic range, for example: lgtBlueToDarkBlue_30 to describe a material clip that changes from light blue to dark blue over 30 frames.

***Fig. 9**. 3ds max Material Editor with FXTheme compliant materials. Note current material is identified by its name as red to orange material clip intended for use in content deselection.*

In order for material assets to be exported, they must be assigned to an object in a 3ds max scene. More details on creating Clips for materials and visibility are provided below in organizing a 3ds max Material and Visibility Asset File.

Visibility Assets

Visibility assets are used to make PowerPoint content appear and disappear smoothly (fade on and fade off). They are always Clips. To create Visibility Assets, add a simple geometry object to your 3ds max scene, then add a visibility track to it. Animate that track from the visibility values you choose and over the time period you choose. In this case it is the object name (vs. the material name) that will be exported and referenced in the FXDesigner. Therefore choose visibility object names like:

- visibilityOn_30 (fades on in 30 frames)
- visibilityOff_120 (fades off in 120 frames)

The special designation of Visibility Assets as Clips is covered below in the section entitled Material and Visibility Clips.

Organizing a 3ds max Material and Visibility Asset File.

As stated, materials must be attached to max scene geometry in order to export. There is no requirement of a rendering camera or light in the scene, as those elements do not export as Assets. As for Visibility Assets, simple geometry such as boxes will suffice for assignment of materials. In the case of Material Assets export, the object name is not critical. The material name is exported, not the object name. In a file with many materials, however, it will be useful to have object names that indicate what material they represent.

Tip: A useful way to organize your Material and Visibility Asset scene file is to spatially separate the three different collections of boxes. These would be for static materials, animated materials (material Clips), and visibility Clips.

An example snapshot of a 3ds max perspective view looking at such a file is provided as Fig 10.

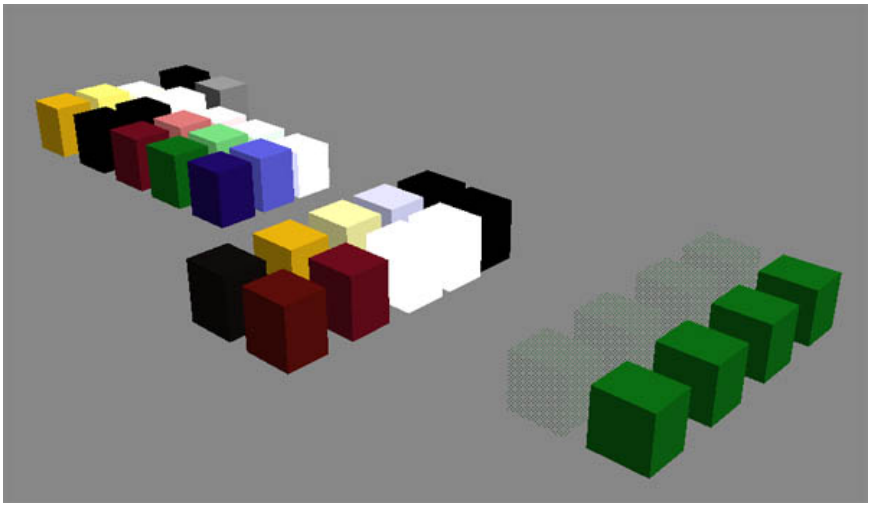


Fig. 10. *A 3ds max scene set up for creation and export of materials and visibility Clips. Visibility objects in lower right. Material clip objects are in the center. Static material objects are in upper left. Max timeline is on frame 1 so this represents initial state of all material and visibility Clips.*

Here are a set of housekeeping tips for with creating a Material and Visibility Assets file.

- Arrange static materials into rows of hues from dark to light
- Segregate select and deselect material clip objects into pairs
- Segregate visibility on / off objects into pairs based on duration of effect

Material and Visibility Clips

For Materials and Visibility Assets to be designated as Clips, two things must occur.

- The appropriate elements must be animated, and
- The objects on which those visibility and material properties have been assigned must be tagged appropriately in 3ds max as Clips

The visibility attribute of our visibility objects has already been animated. If it has not been done already, we now need to animate those aspects of our dynamic materials that we want to change when the material clip is triggered. Basically, any material attribute that is supported in Presenter can be animated. Use the standard 3ds max techniques for creating keyframes on material attributes over a specified time.

Now for step 2. First, select all the visibility objects in the scene. Then open the Properties dialog box in 3ds max and select its User Defined tab. In that tab enter the following string:

Clip=V

This User Defined Property designates the visibility animation curves on the selected objects as visibility Clips for the FXML Exporter. Spacing and capital letters are important. The clip duration will extend from the first visibility keyframe to the last.

Next select all the scene objects to which animated materials have been assigned. Go back into the Properties dialog and again select User Defined. In that tab enter the string

Clip=M

This User Defined Property identifies the material animation curves on the selected objects as material Clips for the FXML Exporter. Again, spacing and capital letters are important, and again the clip duration will be from first keyframe to last keyframe.

Exporting Material and Visibility Assets

Now that the Material and Visibility Assets are created, we're ready to output them using the FXML Exporter. As in exporting Scenes, first call up the FXML Exporter from the 3ds max Utilities panel – see Fig. 2. With the name of the target Theme Pack and (optionally) Theme entered, clicking on the Export Assets button will bring up the Export Assets control dialog. See Fig. 11.

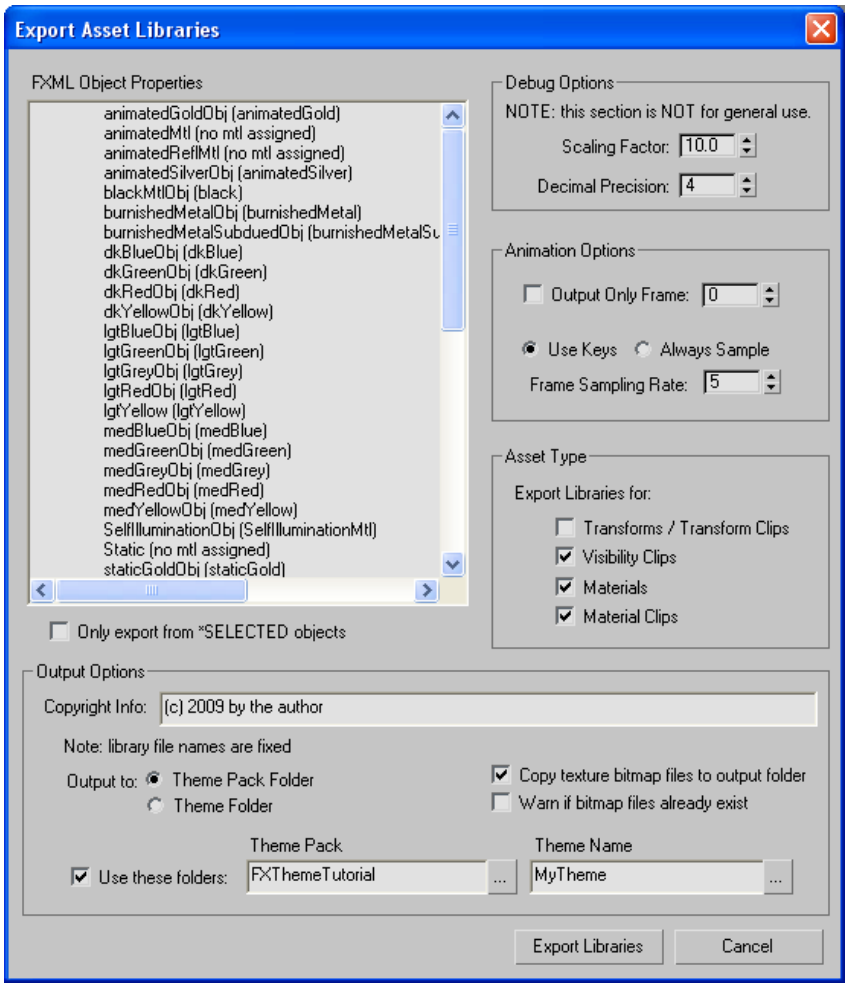


Fig 11. The Export FXML Assets dialog in Instant Effects 3ds max FXML Exporter. Here it is set up to export static Materials, Material Clips and Visibility Clips into an Asset Library FXML file that will be located in the Theme Pack folder

In the upper left, this panel provides a list of 3ds max scene objects and elements that will be exported as FXML assets. Object name, material assignment and clip attributes are listed. In Fig. 11, note that:

- Material clip objects are designated with "M"
- Material objects show attached material name
- Scrolling up in this list would show static Material objects with no clip property
- Visibility clip objects are designated with "V"

Motion Assets

Motion Assets will be used to drive the way PowerPoint content enters and exits a slide during presentation. They also will control what happens when content is selected and deselected by the user. Motion Assets are primarily of the user triggered type of animation called Clips. Those Clips can be applied to the following types of PowerPoint content in creation of an FXTheme:

- Titles and bulleted text
- Drawing shapes
- Picture objects
- Lines and Arrows

Motion is based on animation of position, rotation and scale. These types of motion are commonly called “Transforms” in computer graphics. Both the FXML Exporter and the FXDesigner will refer to the Motion Assets created and referenced in the Theme building process as Transforms and Transform Clips. Transforms describe motion that is continuous and occurs without being triggered by the user. Transform Clips describe motion that is initiated by some specific user input during presentation. Motion Assets are almost always Transform Clips.

Organizing a Motion Assets 3ds max File

We start with a max file that is dedicated to just the creation of Motion Assets. In that file the attributes that will be exported are motion curves, but as with Materials and Visibility Assets, those motion curves have to be attached to scene geometry. Again, the complexity of the geometry does not matter so we choose a simple box. – see Fig. 12.

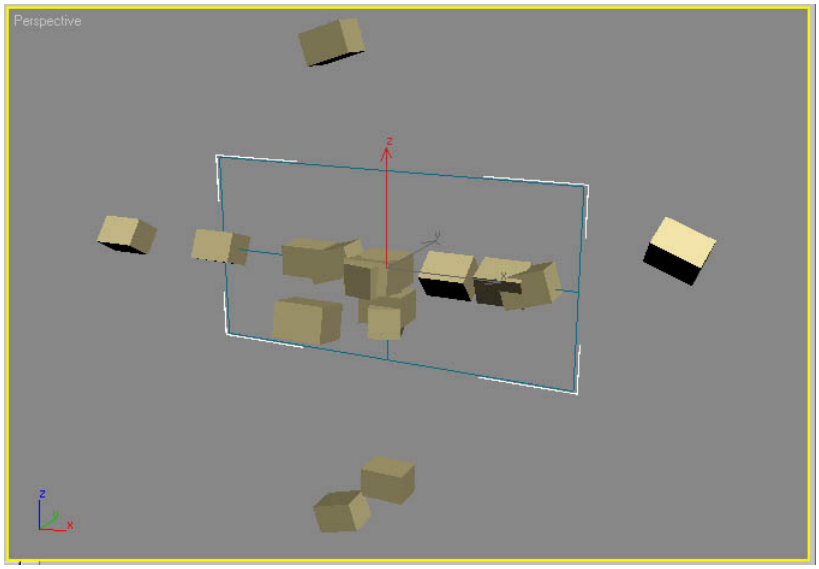


Fig. 12 – *Perspective view of Motion Asset 3ds max file. Origin is at center. Time line is on frame 20 showing position of individual Motion Assets at that time. Note the parent rotation plane.*

Parent Rotation Plane: One complexity of creating Motion Assets for export to the FXDesigner is initial orientation. Because 3ds max operates in a different coordinate system orientation than PowerPoint, we must first create a dummy object and rotate it 90 degrees in X. All our stand-in Motion Asset boxes will be created as children of this Parent Rotation Plane – see Fig 12. Motion stand-in boxes can then be animated so that the 3ds max Front view corresponds to PowerPoint’s view of its content.

Tip: As with the materials and visibility 3ds max Asset Library file, a camera is not strictly necessary for the motion file either, as it is not an exported asset. However, including a camera at the Presenter default position of 0 in X and Z, and -175 in Y, with a FOV of 50 degrees will allow you to judge what the content motion will look like in a presentation, including whether it starts or stops off screen.

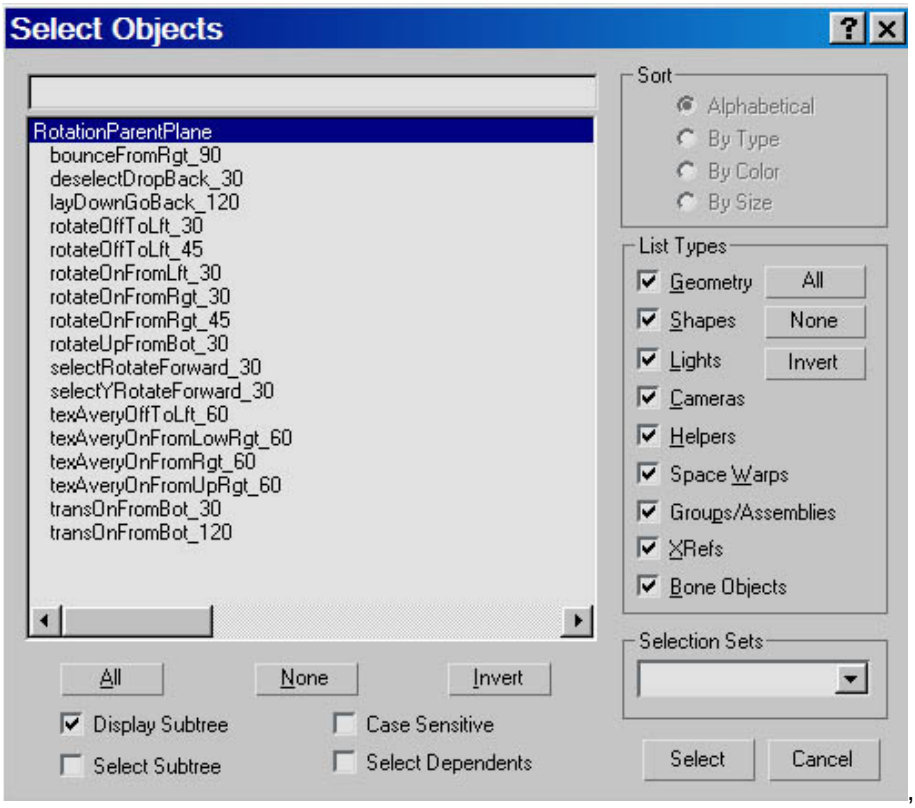


Fig. 13 – 3ds max Select object dialog showing Rotation Parent Plane as parent of all animated motion stand-in boxes.

It is the name of the motion stand-in boxes that gets linked to the actual motion data on export. For that reason make sure to pick names that describe the motion. Examples would be:

- bounceOnFromRight_90 – bounce motion entry animation taking 90 frames (3 seconds)
- rotateOffToLft_30 – rotating exit to left taking 30 frames (1 second)
- selectRotateForward_30 – a selection animation that rotates towards the viewer in 1 second

Motion Asset creation considerations

There are several important considerations in creating Motion Assets.

- If the scale of PowerPoint content is animated with a Motion Asset (Transform clip), the scale value should end at 1.0. Otherwise the content will end up sized differently in Presenter than on the user's slide in PowerPoint. This would reduce the WYSIWYG connection between PowerPoint and Presenter and will make authoring by the end user more difficult.
- Triggered Motion Assets (Transform Clips) created in 3ds max must either start or end at the origin. The motion boxes shown in Fig 12 all either start or stop their animation at a position of 0, 0, 0. An end position at the origin is for Clips that either cause content to enter or deselect. A start position at the origin is used for Clips that cause content to exit or select.
- The origin in a Motion Asset 3ds max file corresponds to the local origin of each piece of PowerPoint content in a presentation. That local origin may be the lower left or upper right of the frame. Motion Assets in 3ds max should cover adequate distance to accommodate this.

Motion Clips

Like Material and Visibility Assets that are intended as triggerable behaviors, Motion Assets that will be initiated by user interaction must be designated as Clips within 3ds max in order for the FXML Exporter to detect them as such. This is done in the same fashion – by including a "Clip=" comment as a User Defined Property for each motion box. Select all the stand-in motion boxes. Open the 3ds max Properties dialog. Select the User Defined tab. On that page type:

Clip=PR

This identifies the motion curves in the position and rotation channels of that object as Clips. Again, capital letters and spacing is critical.

Note: including "S" (as in Clip=PRS) in the user defined property will have no scaling effect in the resulting clip if scale has not been animated.

Motion Asset Export

Once the Motion Assets file is created in the above manner, and motion stand-in boxes have been animated and declared as Clips, it's time to export. Call up the FXML Exporter and click on Export Asset Libraries. You will again get the Export FXML Assets dialog – see Fig. 14.

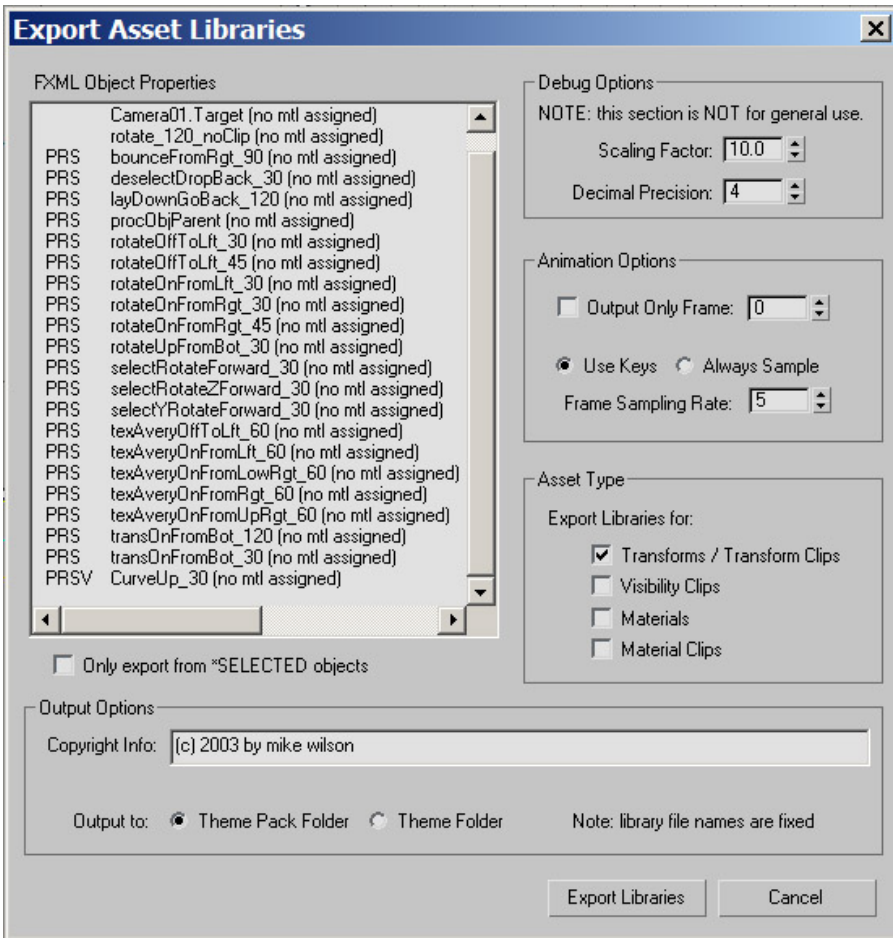


***Fig 14**. – The Export FXML Assts dialog set up for exporting Transforms and Transform Clips.*

The same Theme Pack vs. Theme destination issues pertain. If the Motion Assets are to be used by the entire Theme Pack, the Asset Library file should be exported to the Theme Pack folder. If that file will be unique to a particular Theme, then it should be output to the named Theme folder. The files that will be created in the destination folder are:

- Transform.fxml, and
- TransformClip.fxml

Continuous Material, Visibility, or Motion Animation

There are occasions in Presenter where Assets are intended to animate without triggering by the presenter. Examples would be:

- A reflective material where the reflection changes slowly and constantly
- A 3D bullet which rotates slowly.

This type of cycling animation is much more common with Scenes than Assets, but where it needs to be created, it is accomplished in the following way.

Create the animation desired, whether it is material animation, visibility animation, or motion. Using the 3ds max Track View Graph Editor, bring up the motion curves for that object or material. Add an out-of-range controller to the animated motion curves. This will cause the object to cycle. Finally, do NOT add a “Clip=” user defined property to that object.

On export, Assets created in this manner will cycle continuously when they are used in Presenter.

Tip: Out of range controllers for path animation constraints do not export directly. Path animation constraints can be used to create the motion path for objects, but in order for them to cycle continuously in Presenter, they must first be collapsed to create explicit position information for those objects. To do this, use the 3ds max Collapse Controller feature in Track View. Out of range controllers can then be added to the positional data created by the collapse controller option.

Considerations on FXML Asset Library Export

The following are some key considerations on working efficiently with Asset Libraries. While they have been stated before, they are important enough that we’ll repeat them again here.

- The file names for Asset files are hard coded. Control available to the Theme author is based on where the Asset files are placed. If they are written to the Theme Pack folder, they are available to all Themes created within that Theme Pack. If they are written to a specific Theme, those Assets will only be available to that theme.
- Subsequent exports overwrite previous exports. They are not additive and there is no notification when files are overwritten. For this reason it is good practice to associate a single 3ds max file with a Theme Pack or Theme. When additions or changes are required, they can be made to that file. The subsequent Asset Library export will overwrite the previous version, but only the changes or additions will differ from the previous version.

3D Bullets – A Special Kind of FXTheme Asset

Presenter offers the FXDesigner the ability to include a family of 3D objects bullets as one of the design elements of FXThemes. Within Presenter, these bullets are displayed in the Content scene. They are viewed by the camera of the Content scene and lit by the Content scene’s light(s) and adopt their behaviors from the Theme’s bulleted text controls. This includes entry and exit animation as well as selection and de-selection behavior. 3D bullets may also have their own cycling animation that is taken from the Theme’s (or Theme Pack’s) Transforms.fxml asset library file. Similarly, materials for 3D bullets are taken from the Theme’s (or Theme Pack’s) Materials.fxml asset library file, and any maps referenced by the material are searched for along with other theme materials. In these ways, 3D bullets are treated like PowerPoint content. The details of creating 3D bullets and adding them to FXThemes are provided in the section on Advanced FXTheme Creation Topics.

Tutorials - Building Insert Scenes and FXThemes

Section Overview

This section will introduce the techniques and tools used for building Insert Scenes and full FXThemes for Presenter. It employs a tutorial approach that incorporates a set of example 3ds max files that are provided for this purpose. These files will work with versions of 3ds max back to release 7.

Our first project will be development of Insert Scenes. They are the most straightforward custom assets to create for Presenter. With Insert Scenes covered we will move into the process of building full FXThemes. To construct FXThemes we will use the stand-alone FXDesigner application to organize Scenes and Assets that have been exported from the example 3ds max files. The FXTheme building tutorial starts with the most basic elements of foreground and background scenes, then expands into more advanced topics such as custom transitions, 3D bullets, and even addition of DirectX9 pixel shaders.

To conclude, we will use the stand-alone FXPackager to create secure FXTheme and Insert Scene FXPackage files that can be distributed to other users of Presenter

Example Tutorial Files

The 3ds max files provided for this tutorial are rudimentary, but will be sufficient to illustrate the Insert Scene and FXTheme creation process from scratch. With the basic elements of Presenter content development understood, 3ds max artists will be able to bring their own creativity to the process and generate a wide array of visually stunning FXThemes.

The 3ds max files provided are in the folder called FXThemeMaxFiles that was created when the AuthoringToolkit.zip file was unzipped. It contains 3ds max files as well as bitmap images and a shader example that will be used in these tutorials. The 3ds max files are:

- background.max: To create a subtle animated background scene file for an FXTheme
- logoForeground.max: To create an animated logo scene as a foreground for an FXTheme
- motionsFile.max: To create Asset Library files for transforms and transform Clips
- mtlFile.max: To create Asset Library files for materials, material Clips, and visibility Clips
- cameraLight.max: To create a Main Content scene with animated lights
- insertScene.max: To create an example Insert Scene
- shaderInsertScene.max: To create Insert Scenes that uses custom shaders
- DVE transition.max: To create scenes for a DVE transition
- R2TcontentTransition.max: To create a scene for a Render to Texture type transition
- Twist Deform Transition: To create a Renter to Texture transition that uses deformations.
- contentVariation.max: To create a scene for a theme variation.
- logoForegroundTeapotPour.max: To create a special foreground scene to be used in our Render to Texture transition.
- 3D Bullet.max: To create a 3D bullet for inclusion in our FXTheme.
- MultMaterialExample.max: Illustrating new support for Multi-Materials.
- VideoInsertBillboard.max: An example that creates a Video Insert Scene
- Dragon_Character.max: Use of deformations on an animated character in an Insert Scene.

There are two additional folders provided with supporting content. Those are:

- Media: Containing various bit maps used for textures and reflections in the above 3ds max files plus small FXThemeTutorial.jpg and BasicTheme.jpg files to be used as thumbnail images for the Theme Pack and FXTheme that will be created as part of this tutorial.
- Shaders: Containing the file NewBumpReflect.xml: This is a shader control file for connecting a 3ds max file to a custom shader. That process is also covered in this tutorial

As a first step, move the AuthoringToolkit folder to where you store your 3ds max projects. It is saved in a nearly flat hierarchy with the 3dsmax files at the top next to a single Media folder that contains all the bitmap images listed above.

Insert Scenes

Insert Scenes are easiest of all Presenter content elements to create as they do not require the definition of backgrounds, foregrounds, or PowerPoint content attributes, and they do not need to be assembled in the FXDesigner. Insert Scenes are versatile. They can be used in any presentation. They are NOT linked specifically to any particular FXTheme. Insert Scenes are a great place to start learning how to create custom assets for Presenter.

Insert Scenes are highly useful as a means for representing company logos. Those logos can be nicely animated, lit, shaded, and then applied to any or all slides of a presentation. Animation can be eye catching and emphatic for maximum impact on entry or conclusion slides, or it can be subtle and subdued for Insert Scenes that work well as corner logos.

Insert Scenes can be precisely positioned and sized on a per slide basis. They can be enabled for 3D manipulation by the end user, or that facility can be disabled. This flexibility provides a simple mechanism for adding optimized corporate branding to even generic themes.

Insert Scenes are entirely self-contained. They require complete scene definition, including all geometry, cameras, lights, materials, textures, and animation. Unlike scenes used for creating FXThemes, all media assets for Insert Scenes must be contained in the scene folder that results when they are exported. Bitmaps used for textures and reflections cannot be located in a Theme or Theme Pack media folder because Insert Scenes are not linked to any particular Theme Pack or Theme.

When exporting Insert Scenes from 3ds max using the FXML Exporter utility, the Insert Scene folder is created in your OfficeFX / Repository / Insert folder. The Insert Scene folder gets its name from the Scene Name that is specified in the FXML Exporter utility – See Fig 15.

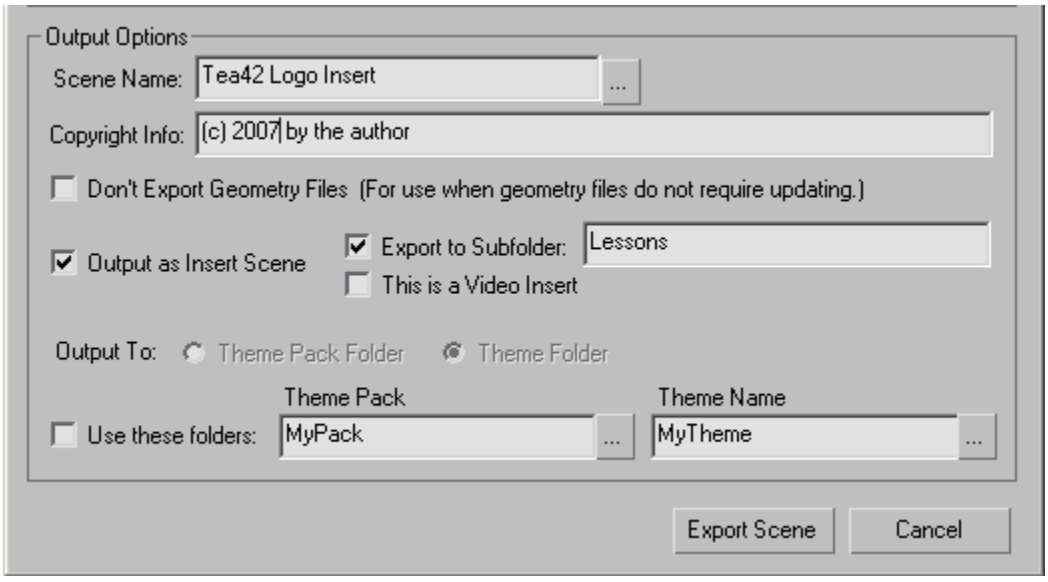


Fig. 15 – Bottom half of the Scene Export dialog showing the "Output as Insert Scene option box selected and the Scene Name filled in. The "Export to Subfolder" option is also checked. This export will create a new folder named "Lessons" in the Office FX / Repository / Insert folder. In the Lessons folder will be a folder named Tea42LogoInsert. That folder will contain a file called Tea 42 Logo Insert.fxml that carries all scene description information as well as binary .x files for scene geometry.

Tip: The Subfolder option is helpful when you are creating a set of Insert Scenes for a specific client or product line. It will help you and your client keep a better organized Insert folder.

A basic example of an Insert Scene is provided as a 3ds max file. It is called insertScene.max. Note its use of two cameras for viewing different geometry from different cameras. After first installing the FXML Export Utility, (see install instructions in the "FXML Exporter Utility for 3ds max" section of this document) load the insertScene.max file into 3ds max and export it as an Insert Scene. The file uses a bitmap called METAL7_smsq.jpg as a reflection for one of the teapots. Make sure that bitmap is found by 3ds max. The FXML Exporter will then include that bitmap in the new Insert Scene folder it creates in OfficeFX\Repository\Insert. Now, start up Presenter and load the new Tea42 Logo Insert Scene. Note that you have two cameras to choose from (using the "Versions" pulldown in Presenter's Insert control interface) and that you can load this Insert Scene into any slide of any presentation, using any FXTheme.

Insert Scene Viewport Aspect Ratio

Viewport aspect ratio is a critical consideration for all scenes created for use with Presenter. The viewport aspect ratio for all Presenter scenes is controlled by the aspect ratio of the camera that's used to view the scene in 3ds max. This is important for Insert Scenes because their viewport cannot overlap the edge of the full Presenter display window.

The default camera viewport in 3ds max is 640x480 (for an aspect ratio of 4:3). If this default aspect ratio is used for Insert Scenes where the content has a "natural aspect ratio" that is much different than 4:3, the user of those Scenes will not be able to position them on their slides with complete flexibility.

As a practical example, the text only version of the Instant Effects logo is much wider than it is tall. When exported as an Insert Scene with the 3ds max default camera of 640x480, that Insert Scene cannot be positioned at the bottom of a slide and still have the logo itself be reasonably sized. On the other hand, if the camera viewport in 3ds max is flattened out to match the "natural aspect ratio" of the Instant Effects logo, the Insert Scene that results can be sized and positioned much more flexibly in Presenter – See Fig 16.

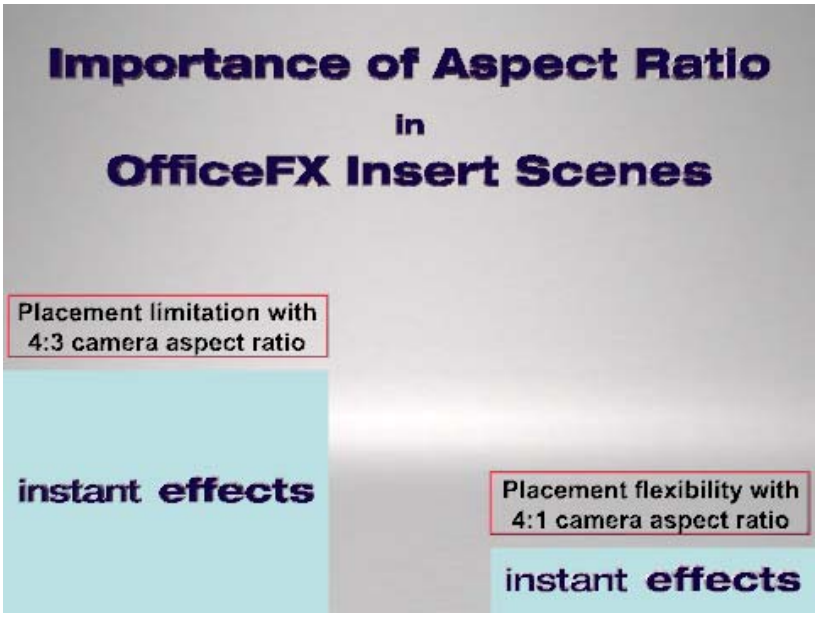


***Fig 16**. – In the lower left example, the Instant Effects logo is exported with a 4;3 aspect ratio camera. The light blue rectangle shows its 3D viewport. That viewport cannot overlap the Presenter frame so the logo cannot be pushed to the bottom of the slide.*

When the same file is exported with the camera in 3ds max set to a 4:1 aspect ratio as in the lower right example, the resulting Insert Scene has more layout versatility.

Tip: When setting up 3ds max cameras to control 3D viewport aspect ratio, the absolute resolution is not important. It's the ratio between horizontal and vertical resolution that is key. For example an Insert Scene that uses a camera with a resolution of 400 x 300 will generate exactly the same result in Presenter as a scene whose camera resolution is set to 1200 x 900. Both would have the same 3D viewport aspect ratio.

There are additional considerations and capabilities associated specifically with the creation of Insert Scenes that should be observed in order to get best value from those assets.

- The scene camera(s) used in building an Insert Scene should always be Target Cameras. This is because the camera interaction that Presenter supports uses the camera's target as its center of navigation. If the user in Presenter interactively rotates his or her view of the Insert, the camera will always point at the target. In general the camera target should be placed at the center of the scene geometry which you will want the user to view.
- The user interaction provided in Presenter for viewing Insert Scenes actually moves the camera rather than the scene geometry. This means the user will be free to look at objects from behind since scene lights do not move with the camera. You will want to consider lighting Insert Scenes so that key geometry is lit from all possible camera positions.
- Multiple cameras can be included. These can be used to present different initial views of scene geometry or to view entirely different scene geometry. When multiple cameras are included in an Insert Scene, they are available for selection in the Presenter Insert user interface from the "Versions" pull down.
- Normally cameras for Insert Scenes should not be animated as such animation conflicts with interactive input from the user. When this occurs the symptom is uncontrolled camera motion in Presenter.
- As in other FXML scene files, instancing is supported in Insert Scenes. If your goal is to show multiple material options on a product, use instancing and multiple cameras to save on file size.

Tip: In the case where multiple scene cameras are used to view <u>different</u> scene geometry, you will want to set the two different geometries an adequate distance apart and adjust their viewing camera's far clipping planes so that one camera cannot be moved to a place where both objects will be visible.

- Cycling animation is supported in Insert Scenes. Use out of range controllers in 3ds max to create animation that cycles in Presenter Insert Scenes. Both the lights and the chrome teapot in the example 3ds max scene provided employ cycling animation.
- In general, clips are not used to trigger animation in Insert Scenes (exceptions are covered in the Advanced Topics section of this document entitled "Triggeing Insert Scenes with Action Sets"). Most commonly, Insert Scene animation is set to start with a user click, on slide display, or to cycle continuously in Presenter
- Upon completing an Insert Scene, you can optionally combine it with others in the same folder to make efficient use of shared assets. The folder's name is merely an indicator of what's within, so you can create a "library" folder of similar Insert Scenes, and the user can choose which to load from within Presenter.
- Presenter recognizes only the Standard materials of 3ds max. Files that use other material types such as "Architectural", will need to be converted to Standard materials before they will export and display properly in Presenter.

*Tip: When building a custom Insert Scene for use by others, it is good practice to provide a default Standin Image – even though users can easily build their own in Presenter. This assures that a representative Standin will be placed in the PowerPoint file as soon as the Insert Scene is used. To do this, most easily use the Custom Standin utility from within Presenter's FXEdit / Insert control dialog. That will place an image named **InstantEffectsStandin.png** in your MyDocuments folder. If you rename that to **YourInsertSceneName.png** and put it into the **YourInsertSceneName** folder that was created when you exported your Insert Scene, that PNG file will become the default Standin Image for that new asset.*

Insert Scenes provide a straightforward and powerful way to add 3D content to any Instant Effects presentation. They can be used to amplify branding with dynamic company logos as well as to show off product appearance and functionality. Experiment with modifications to the provided 3ds max file to gain a broader understanding of how Insert Scenes work. With that knowledge as a foundation, we can move on to the creation of complete FXThemes using the Instant Effects FXDesigner.

Building and Packaging FXThemes with FXDesigner

Building full FXThemes is where the doors to creativity open wide. Instant Effects provides a specific application for this purpose called FXDesigner. FXDesigner is a stand-alone Windows application where FXML scenes and assets are assembled into FXThemes. It works through straightforward list-picking where different scenes and assets are assigned to different roles in the theme.

Also provided with this Authoring toolkit is the FXPackager. The FXPackager bundles the FXThemes and Insert Scenes you create into an encrypted form called an "FXAssets" file that can be sent to other users of Presenter. FXThemes and Insert scenes "packaged" with this utility are installed on the recipient's computer using the Products / Install FXAssets feature on the Presenter toolbar.

Note: In the event you are sending both custom Presenter assets (FXThemes, Insert Scenes) AND a finished PowerPoint file that uses them, it is recommended that you instead use the Publish FXPresentation feature from within Presenter itself.

Installing FXDesigner and FXPackager

Both the FXDesigner and FXPackager can be run as stand-alones, but both do rely on files that exist within the Presenter application file system. Installing FXDesigner and FXPackager does not require running a setup program, but both executable programs must be located in the Instant Effects\OfficeFX file folder in order to run properly. Both these programs only run alongside a fully licensed copy of Presenter. FXDesigner will run in trial mode for 30 days before it requires purchase and licensing.

When you unzipped the AuthoringToolkit.zip file, the folder called Executables was created. You've already installed the appropriate FXML Exporter utility from that folder into your 3ds max plugins directory. In the Executables folder you will also find:

- FXPackager: for packaging up FXThemes and Insert Scenes for delivery to other users (FXPackager.exe)
- FXDesigner: for combining FXML scenes and assets into finished FXThemes. (FXDesignerPPT.exe)

Take a copy of both FXDesignerPPT.exe and FXPackager.exe and place those in your Instant Effects\OfficeFX directory. From this location, both applications will have access to other Presenter utilities that are required for accessing Theme and Theme Pack asset folders. Again note that a fully licensed version of Presenter is required to run both these programs.

With FXDesigner properly installed, it can be called from the FXML Exporter utility in 3ds max. It can be started by double clicking on the application in the OfficeFX folder, or it can be started from a shortcut made from the FXDesignerPPT.exe file. The shortcut can be placed anywhere. When you're working on Presenter asset creation, it's often convenient to have the shortcut right on your desktop.

Creating a Theme Pack and FXTheme

With the FXDesigner installed, our first step in this tutorial will be the creation of a new Theme Pack and FXTheme. To do this, bring up the FXDesigner. Go to File / New and select Theme (Fig 17.)

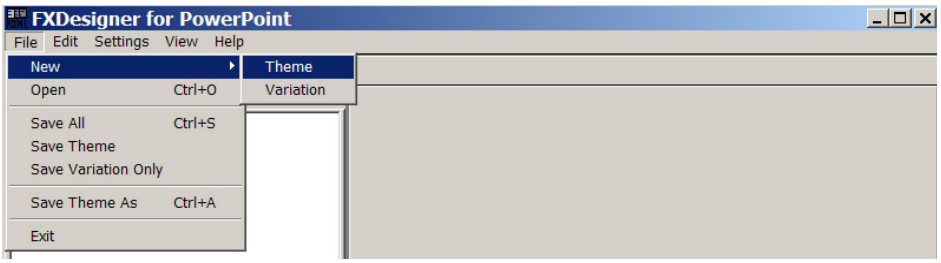


Fig. 17 – File New for creating both new Theme Pack and newFXTheme. This will bring up the Theme / Variation Setup dialog (Fig. 18)

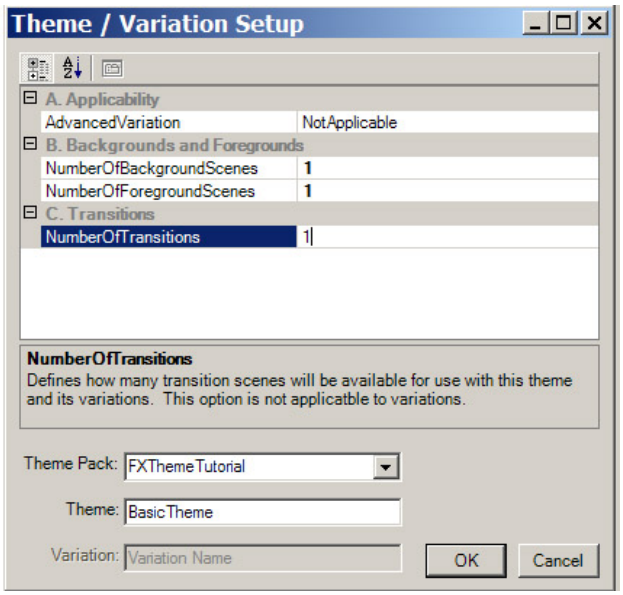


Fig 18. The Theme / Variation dialog enables first configuration and naming of Theme Packs, Themes, and Variations. It also allows declaration of number of backgrounds, foregrounds, and transitions. Those numbers can be changed at a later time

The user can select from an existing Theme Pack or create a new one when adding a new FXTheme. For our first theme, we will start with basics. Choose 1 for each of the Theme attributes:

- NumberOfBackgroundScenes
- NumberOfForegroundScenes
- NumberOfTransitions (Note: this value must be entered by hand)

This tutorial will assume Theme Pack and Theme names as listed in Fig 18. Note: there are no spaces in these names. That is not a requirement. It is just a choice of the author. Names are:

- Theme Pack: FXThemeTutorial
- Theme: BasicTheme

With this data entered save the new theme with File / Save All, then exit the FXDesigner. You now have a useable theme. It is driven only by default assets. Just neutral grey tones are used, but if you start up Presenter now, you will find that your new Theme Pack and theme are already available. Before doing that, we'll place thumbnail images so that the new Theme Pack and theme can be picked visually in both Presenter and in FXDesigner

Thumbnails

You'll make your own thumbnail images for Theme Packs and themes soon enough. Just to get started, you can use the images provided in the FXThemeMaxFiles / Media folder (See Figs. 19 A and 19 B below)

Fig 19 A: FXThemeTutorial.jpg thumbnail for identifying the tutorial Theme Pack.

Fig 19 B: BasicTheme.jpg for identifying the tutorial theme

Copy the FXThemeTutorial.jpg image to your new Theme Pack folder, which should be located here:

Instant Effects\OfficeFX\Repository\PowerPoint\FXThemeTutorial

In order for the thumbnail image to show up in Presenter and FXDesigner, it must be named the same as the Theme Pack. As long as you named your Theme Pack FXThemeTutorial, the supplied image will work.

Copy the BasicTheme.jpg into your new BasicTheme folder, which should located here:

Instant Effects\OfficeFX\Repository\PowerPoint\FXThemeTutorial\BasicTheme

Again, if the theme's folder name is BasicTheme, the supplied image will show up in both Presenter and FXDesigner. You may at this point want to start up PowerPoint, make a couple dummy slides, go into Presenter, and verify that your new Theme Pack and theme are available. If you select the BasicTheme, you'll see that it already works, but it generates nothing more than grey 3D text from your PowerPoint content against a grey background – because we have not yet provided those definitions.

Names in FXML Exporter

In order to create the look and behavior that will be unique to our theme we'll next start creating assets and scenes in 3ds max and exporting those with the FXML Exporter utility. As you bring up the FXML Export utility, make sure to enter the proper Theme Pack and Theme names as this will assure that the exported files are placed into the new Theme Pack and theme folders you've created with the FXDesigner. See Fig. 20.

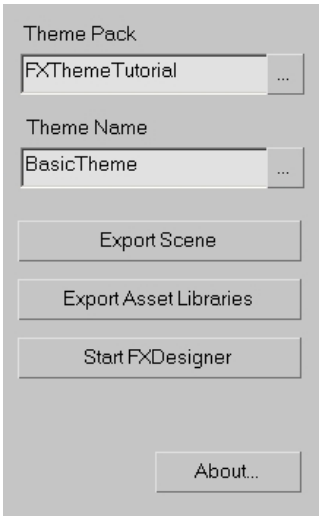


Fig. 20 – Initial FXML Export utility dialog showing Theme Pack and theme names that are recommended for use in this tutorial. These names are used for exporting both Scenes and Asset Libraries. These output names are set in the 3ds max files provided for these tutorials.

Note: Spacing is significant in Theme Pack and FXTheme names. In this tutorial the author has chosen theme pack and FXTheme names that use inner caps and no spaces.

Creating Asset Libraries from 3ds max

First we'll create the asset files that will allow you to control the look and behavior of your PowerPoint . To do this we'll sequentially open two of the 3ds max files supplied with this tutorial and export them.

mtlFile.max

This file will create material and visibility assets. It is organized in the manner documented in the section of this document entitled *Creating Asset Libraries for FXThemes.* Note the user defined properties for materials and visibility Clips. Export in the configuration noted in Fig 11 with Material Clips, Materials, and Visibility Clips Asset Types checked and Output directed to the Theme Pack Folder. The files created will be (respectively):

- MaterialClips.fxml
- Materials.fxml
- FloatClips.fxml

motionsFile.max

This file will create motion assets for the Theme Pack. It, too, is organized in the manner documented in the section of this document entitled *Creating Asset Libraries for FXThemes.* Note the user defined properties for transforms and transform Clips. Export in the configuration noted in Fig 14 with Transforms and Transform Clips Asset Types checked and Output directed to the Theme Pack Folder. The files created will be (respectively):

- Transforms.fxml
- TransformClips.fxml

Creating Scenes from 3ds max files

With Asset Libraries created at the Theme Pack level we switch to creating scenes. These will be specific to (and exported to) the BasicTheme folder

Background.max

Open background.max and browse through. Look in particular at the motion curves for the 3 background circular tubes, noting their out of range controllers that create continuous (cyclical) motion. Then export as a scene named “background” and specify export to the Theme folder (Fig. 21)

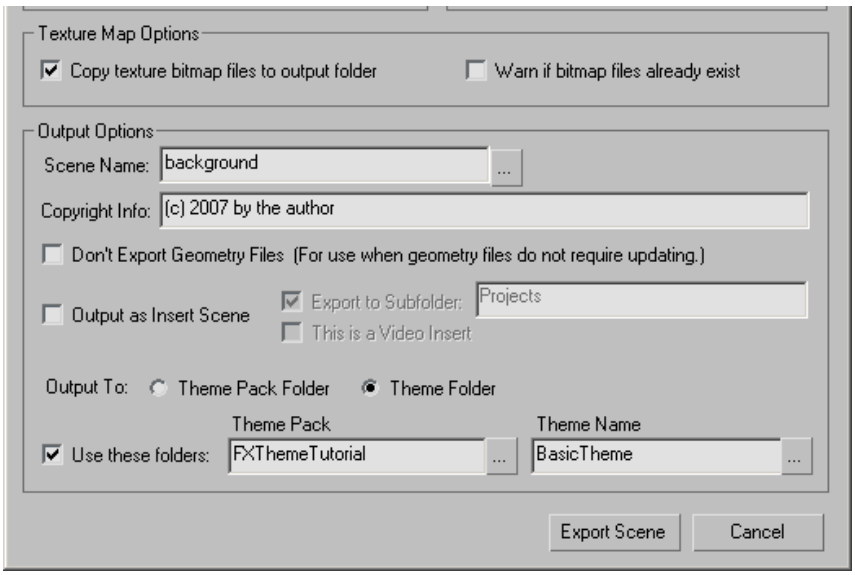


***Fig**. 21 The lower portion of the FXML Exporter utility scene export panel configured to export a scene named “background” to the Theme Folder.*

***Note**: Texture Map Options are set to gather and output the bitmap files that are used.*

logoForeground.max

Open logoForeground.max and browse through. Look in particular at parenting hierarchy and motion curves for the teapot. Note the out-of-range controller on rotation of the TeapotParent dummy object and the separate “Clip=P” user defined property for the translation of the Teapot itself. As a result of this setup, the rotation animation will cycle continuously. The position animation, being noted as a Clip, will only occur when it is “triggered” during transition. We will get to transitions further along in this tutorial. Once you are comfortable with the setup of this 3ds max file, export it as a scene named “foreground” to the BasicTheme folder. See Fig 22.

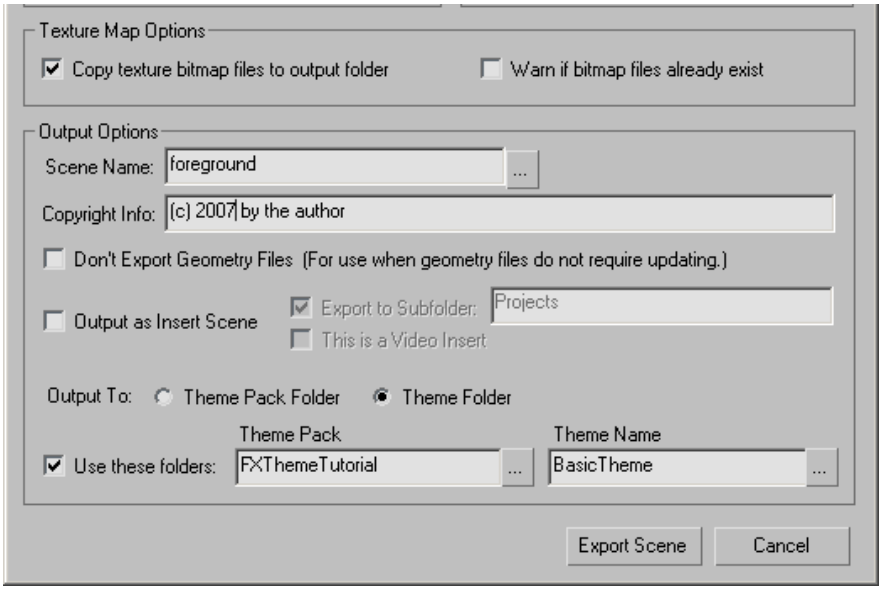


***Fig**. 22 The lower portion of the FXML Exporter utility scene export panel configured to export a scene named “foreground” to the Theme Folder.*

With Asset Libraries and scenes created we can now go back into FXDesigner and begin to expand the BasicTheme. Start up FXDesigner. This time go to File / Open and bring up the Theme Browser (Fig. 23). If you've placed the 2 .jpg images in the target Theme Pack and Theme folders, this time you'll see those thumbnail images in the Theme Browser. You select them just as you do in Presenter. Select BasicTheme.

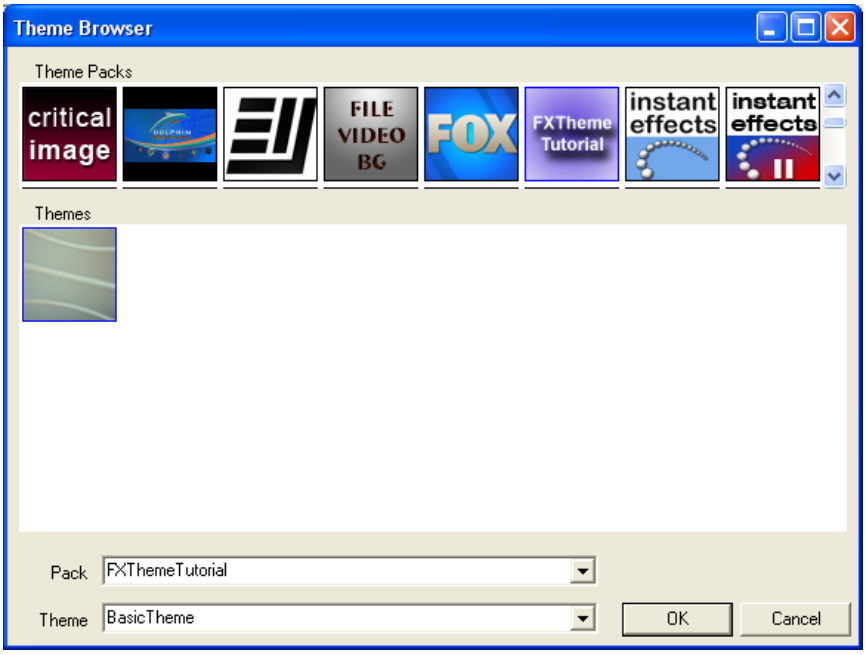


***Fig**. 23 FXDesigner's Theme Browser. Note the new FXTheme Tutorial Theme Pack and BasicTheme Theme thumbnail images.*

Note: If FXDesigner was left running and BasicTheme was loaded while you were exporting the foreground and background scenes, then FXDesigner will need to be "refreshed" in order to know that those new scenes are available. To refresh FXDesigner, click on the "View" button in its top menu bar and pull down to select Refresh Content.

As a first task, we'll assign the background.fxml scene that we just exported to the BasicTheme folder. To do that, expand the theme by clicking on the plus sign next to Theme: BasicTheme in the left panel. Then click on Background 1. Now, in the right panel, click on FXMLSceneName to highlight that option, then click on the file selection down arrow at the far right. A dialog will pop up that lists the eligible scenes. Along with System Default, Theme Default and None, you'll see the two scenes we've just exported – background and foreground. Select the background scene (see Fig. 24).

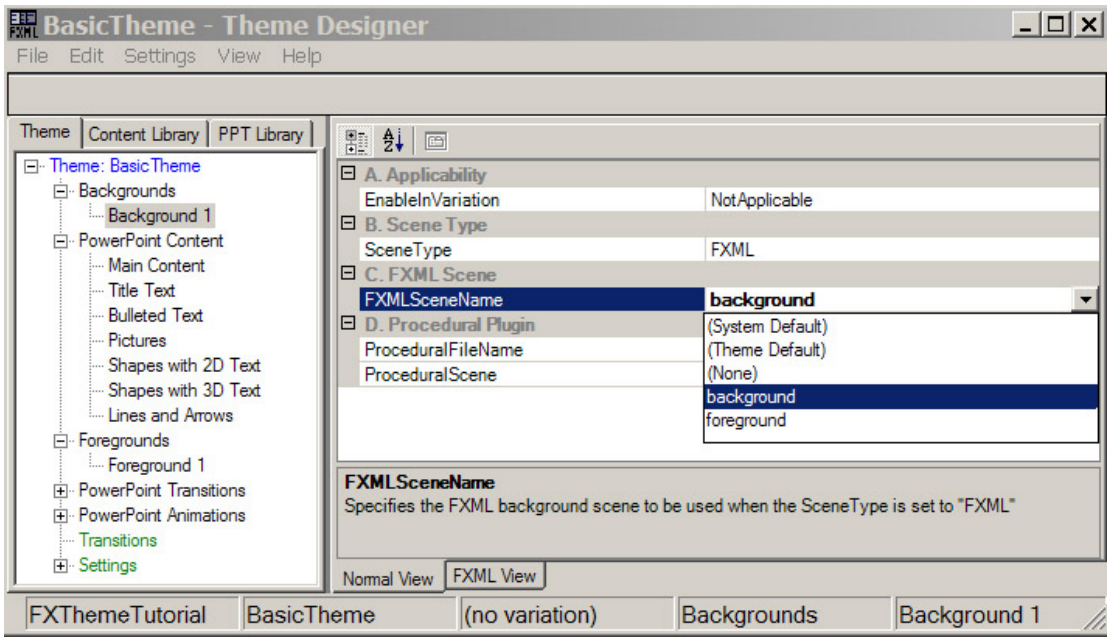


Fig 24 – FXDesigner configured to select a scene as Background 1

Now select Foreground 1 in the left panel, select FXMLSceneName and use the far right down arrow to bring up the FXML scene list. This time assign the scene called foreground (see Fig. 25)

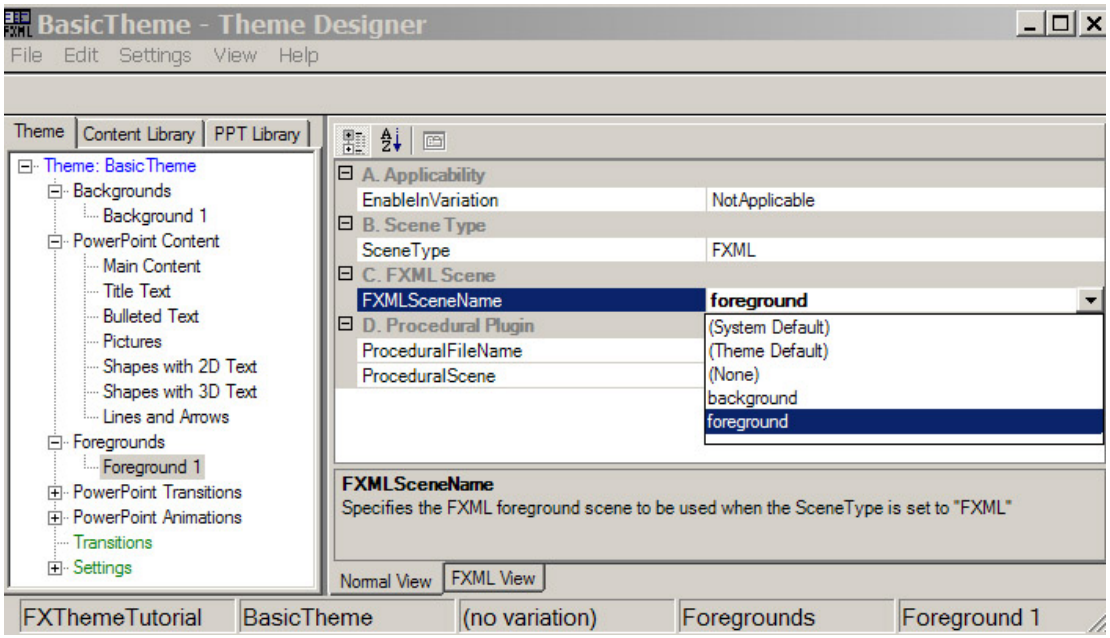


Fig 25 – FXDesigner configured to select a scene as Foreground 1

Go to the File option and Save the theme. If you now go back into PowerPoint and reload your BasicTheme, you'll see the 3 slowly rotating tubes as your background and the "Tea 4/2" logo scene we exported as your foreground.

Theme Treatment of PowerPoint Content

Next we'll use the Asset Libraries to pick materials and animation for our PowerPoint content. For this first pass, we'll just make selections for Text Appearance. As we begin to tune the appearance of our theme, we'll also have to be conscious of our media assets.

Media Assets

Media assets for an FXTheme can be both audio files that are used as part of various effects, and bitmap images that are used as part of materials in the theme. Media assets are searched for in the same order as other theme assets, with one exception. They can also be in their own Media folder, at each level of the search path. For complex themes with many media assets, using a media sub folder (in the theme folder) makes the project more manageable.

As with other types of assets, the search order is from specific to general. First the theme folder is searched, then the Theme Pack, then the Defaults area, so media assets located in the theme folder will only be available to that theme. As has been previously stated, the Defaults area should not be tampered with as that could alter the way all themes function, not just the themes you develop. Best practice is to keep media files for custom FXThemes and Theme Packs stored within those theme and Theme Pack folders.

The FXML Exporter Utility creates a Media sub folder within the Theme Pack and the theme folder when it exports media assets. If you look in the new *FXThemeTutorial* Theme Pack folder you'll see the Media folder that was created when we exported the motion and materials asset files. We explicitly exported those to the Theme Pack folder level. Similarly, if you look in the *FXThemeTutorial/BasicTheme* folder, you'll see the Media folder that was created when we exported the background.fxml and foreground.fxml scenes. Looking in each of those folders you'll see the image files that are used in the texture definitions of materials in those scenes.

If you view the BasicTheme in Presenter you'll see the impact of those materials. For example there is a reflection on the teapot and subtle noise on the background. At the same time, any text you add still shows as plain (default) grey. We'll address that next using the materials and textures that were exported as Asset Files.

Text Appearance

Now we're ready to assign treatment to our PowerPoint text. In the left column, choose Title Text. Choose a Bevel Material, Face Material, and Side Material. The materials list you see is from the Materials.fxml file Asset Library file you exported previously (see Fig. 26).

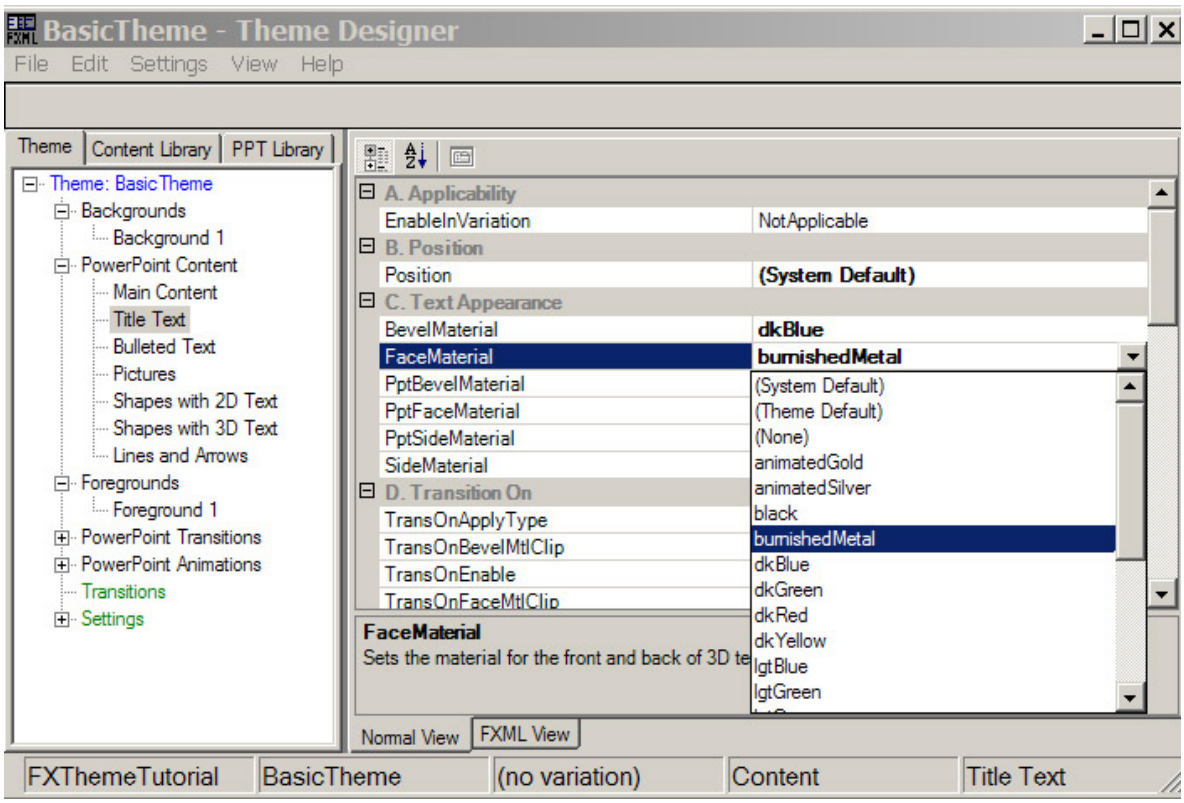


Fig. 26 – FXDesigner showing application of burnishedMetal as Face Material for Title Text.

Make similar selections of materials for Bulleted Text. Remember that the background scene we've created for this theme is primarily light in color. To get decent contrast between your PowerPoint text and the theme's background, you will want to use darker FaceMaterial text colors. When you've made these selections, save the theme. Go back into PowerPoint and re-enter Presenter. If you've not exited, you should still be looking at your few test slides with BasicTheme applied. You now should see the results of your text material assignments. Consider what you would change to improve it, either by selecting a different material from the assets available using FXDesigner, or by opening mtlFile.max in 3ds max, adding a new material, and re-exporting the material.fxml asset library file.

Tips: a) Not all Text options in FXDesigner need to be set. Initially just set FaceMaterial, SideMaterial, and BevelMaterial. b) To inspect your text in Presenter more closely, set the Prefs / Advanced Option to Rotate PPT Content with Right Mouse. c) While running Presenter in Full Screen display, press F8 to see everything in 3D wireframe. Pressing F8 again will toggle back to normal shaded display mode.

Text Behavior

Next, let's tune the behavior of text. One of the capabilities of Presenter that is different from PowerPoint is the ability to create non-linear and interactive slides that respond to user input. A useful way to apply this capability is to add selection and de-selection animations to that content. Typically these animations are applied to bullet points on a slide as these are the points that a user may want to emphasize during presentation.

To apply selection and de-selection actions to your bulleted text in Basic Theme, select Bulleted Text in the left hand column of FXDesigner. Then scroll down to the section labeled F: Selection in the right hand column. You'll see a number of options for adding selection animation. As a first step we will apply just a Transform Clip and make it happen per character (vs. per word or per paragraph). See Fig. 27.

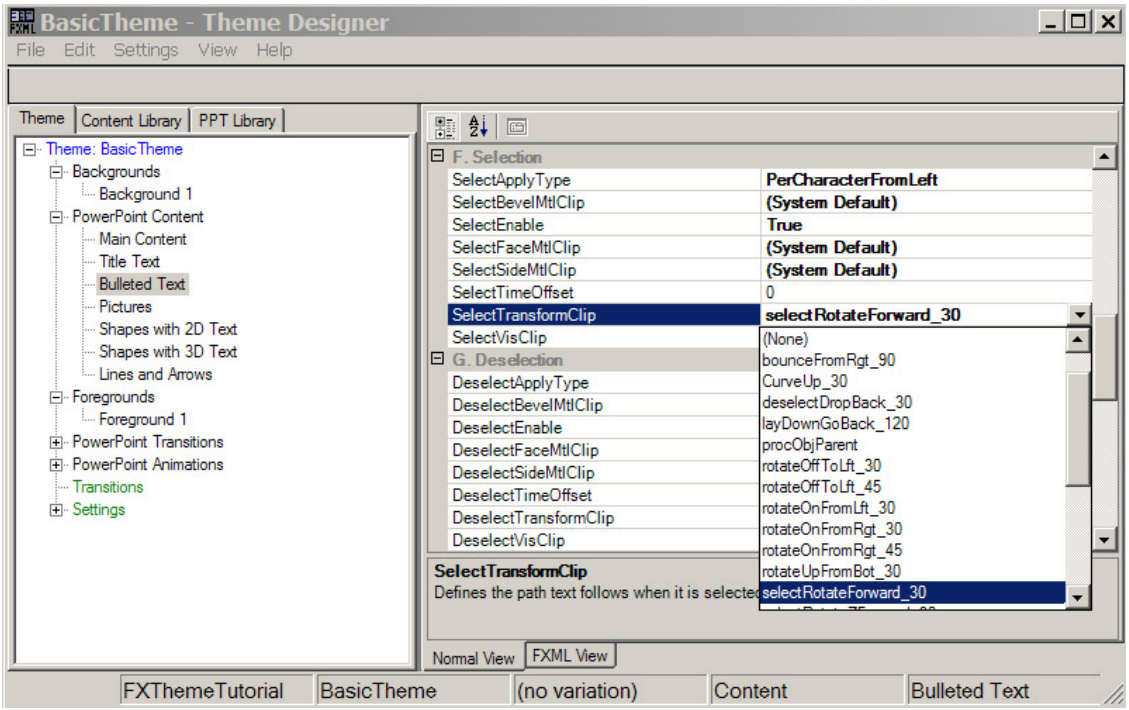


Fig. 27 – FXDesigner showing application of the motion Clip "selectRotateForward_30" to Bulleted Text. The SelectApplyType is "PerCharacterFromLeft". This will apply the 30 frame rotate forward motion clip to each letter of the bulleted text item individually.

Make the selections indicated in Fig 27. Note that the SelectEnable option must be true. The FXDesigner will query if you want to "Enable Select support" if SelectEnable is set to false and you attempt to make choices for the selection attributes. If you answer "yes" to this query, SelectEnable will be set to true. If we leave the SelectTimeOffset at 0, Presenter will apply an internal algorithm that generates an offset between the initiation of each letter's selection animation that adjusts based on the number of letters in the bulleted text entry. This prevents selection actions from taking too much time to complete.

Now go back into PowerPoint, hit Refresh, re-enter Presenter, and evaluate both the appearance and behavior of your PowerPoint text content. Again note that specific settings do not need to be made for every option. There are system defaults for everything. You'll only need to change what is appropriate for the FXTheme you are creating, and in many cases no text behaviors are added at all.

Pictures - Appearance and Behavior

Creating a 3D border and enabling selection and de-selection animations are two basic theme options for displaying images in Presenter. You'll see other attributes of pictures that can be controlled by Presenter indicated in FXDesigner. Those are more advanced FXTheme capabilities.

Select the "Pictures" label in the left column of FXDesigner. You will see the list of Picture attributes that an FXTheme can control. In the appearance, choose a Line Material. This will be the material applied to the 3D border the user selects in Presenter. The System Default Fill Material will suffice. It is a 100% self illuminated white that allows the user's PowerPoint images to be displayed in Presenter without any shading calculation. That assures that their colors are fully saturated.

If you want the theme to offer interaction capabilities with pictures you can choose Selection and De-selection and choose Transform Clips as you did for text.

Note: In the motionsFile.max file provided, selection animation Clips are named specifically. They all start with the word "select", and they all animate slightly forward as well as moving in some other way. They all are 30 frame Clips as well.

For now, don't assign other selection and de-selection attributes such as material clip animations, but do note the more advanced options that are available. When concluded, your Pictures dialog attribute selections should look something like Fig. 28.

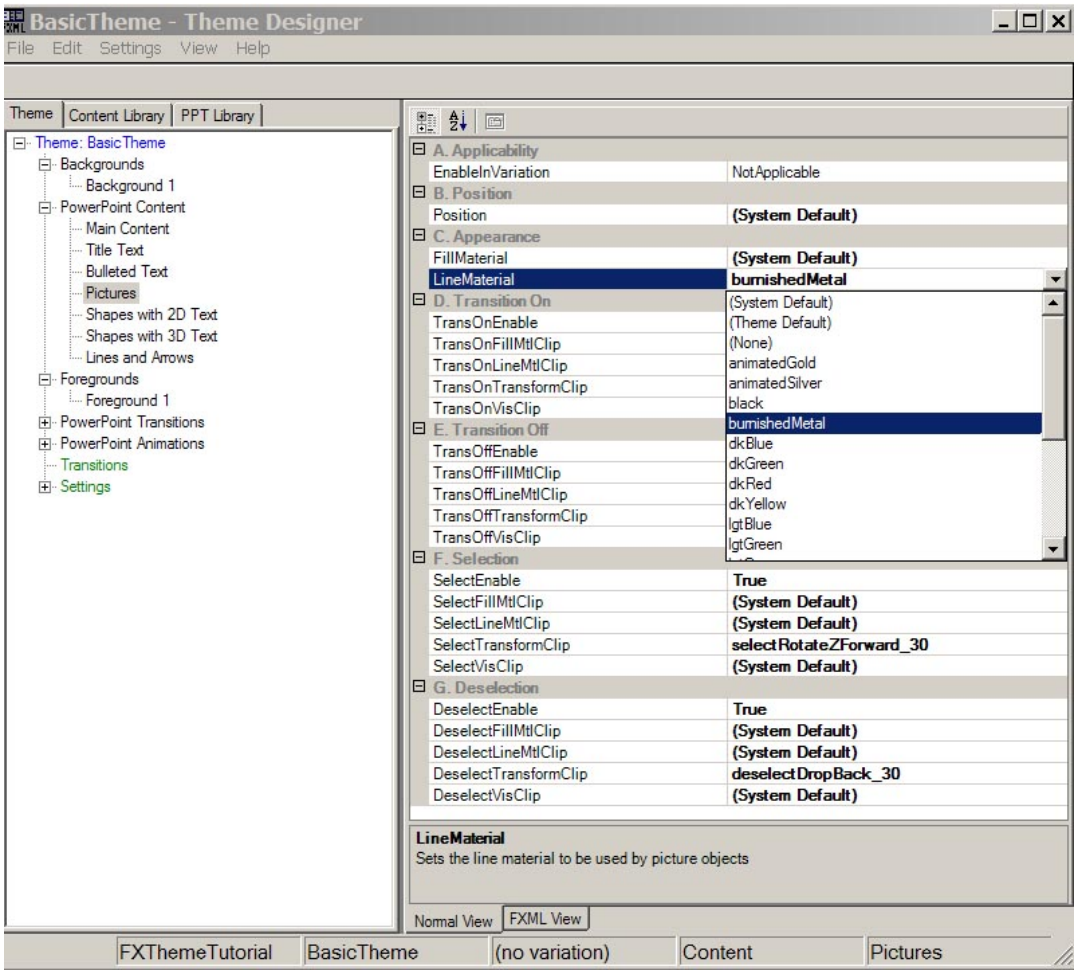


Fig. 28 – Basic Theme Pictures attribute selection.

Shapes with 2D Text – Appearance and Behavior

Shapes are the PowerPoint drawing objects like rectangles, ovals, block arrows, etc. Shapes can be a plain color, can have fill effects and borders, and also can include internal text. Themes can control the appearance and display style of all these elements. It can control behavior of shapes when they are animated or selected by the user. More advanced themes can also control the manner in which shapes are animated onto and off of a slide.

Like Pictures, Shapes can have a 3D border. FXDesigner allows the theme to determine the color of the border with the Line Material attribute. Select that just as you did for Pictures.

The Fill Material for Shapes will generally be left as System Default. This will employ a fully self illuminated base material that allows the color selected by the PowerPoint user to be displayed with no 3D shading. If you do wish to provide a way for you FXTheme to override the user's color selection for Shapes, the FXDesigner allows you to specify a Fill Material. The material you specify as a Fill Material for shapes in FXDesigner will drive all the material attributes for shapes except for diffuse color. The diffuse color for shapes will come from the color or fill effect for the shape that the user generated in PowerPoint.

Note: In order to access the FXTheme's custom shape material color, the user will need to set the Options / Colors option for Shape Fill to Theme instead of the default PowerPoint control.

Declaring a reflective material will cause shapes to exhibit a reflective surface, but that surface will be tinted with the shape's color from PowerPoint. If you wish the shapes fill color or fill effect to appear just as it does in PowerPoint -- that is, fully saturated and with no lighting, shading, or reflective effects – then specify a Fill Material that has been created in 3ds max with 100% white self illumination. In the example material file supplied, the material of this type is called SelfIlluminationMtl.

For shapes with 2D text you can also specify a color for the text. This is done in FXDesigner with an RGB value. For the most part, this color will not be used as users will generally want the text color they've chosen in PowerPoint to determine the color of the text in their shapes. That is one of the Slide Options that Presenter provides

The only other attribute we'll assign to shapes with 2D text in a basic theme is a selection and a de-selection transform. That is done just as it was for Bulleted Text. The motion assets that are appropriate for selection and de-selection in the 3ds max files provided are indicated with names that start with select and deselect.

Shapes with 3D Text – Appearance and Behavior

In Presenter, another option for shapes is for their text to be represented in 3D. You'll see a separate selection in the left column of FXDesigner for selecting the attributes of Shapes with 3D Text. On clicking that entry, you'll see that Shapes with 3D Text share the same Fill Material and Line Materials of Shapes with 2D Text. All other attributes are different, and Shapes with 3D Text attributes include the normal 3D Text Appearance attributes of Bevel, Face, and Side materials. Make selections for those attributes just as you have done for Title Text and Bulleted Text. Most users will wish to color their shape text in PowerPoint. To enable that, make sure to apply SelfIlluminationMtl to the PPTFaceMaterial attribute in the Text Appearance section. If you wish interaction enabled for shapes, make choices as well for Selection and Deselection transform clips. For finished selection of attributes for Shapes with 3D Text, see Fig. 29.

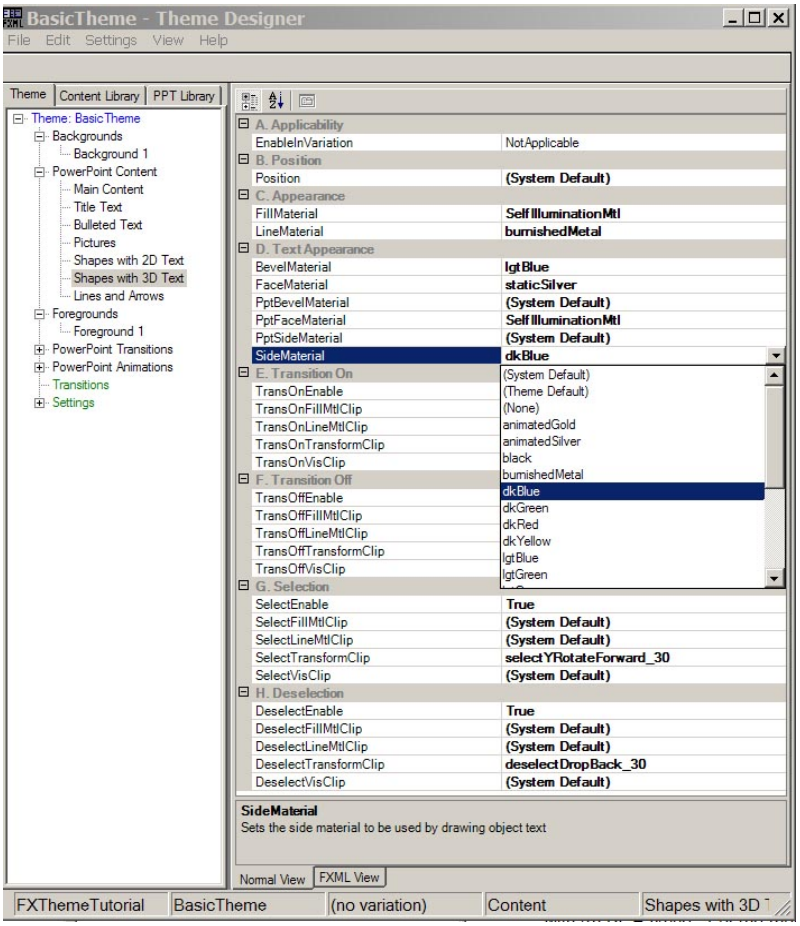


Fig. 29 – FXDesigner attribute selection for Shapes with 3D Text

An additional note on text Appearance that applies here, as well as to Title and Bulleted Text, is that the theme can provide materials that augment or exactly duplicate the color choices a user makes for text in PowerPoint. The Text Appearance attributes of PPT Bevel Material, PPT Face Material, and PPT Side Material accomplish this. As before, a 100% self illuminated white material applied here will cause Presenter to take exactly the color specified by the user in PowerPoint and display that with no shading calculation. For the Basic Theme we will select the SelfIlluminationMtl for PPT Face Material.

Lines and Arrows – Appearance and Behavior

Lines and arrows are simple cases for basic themes. They are assigned an Appearance attribute and can also have selection and de-selection attributes. Often a user will color lines and arrows in a PowerPoint presentation for a particular purpose, for example to represent a particular type of data flow, or a characteristic type of link. In order to enable that color to be represented in Presenter without a shading calculation that could alter the user's selected color, assign the SelfIlluminationMtl as before.

You can also make selection and de-selection transform choices as you have done with other types of PowerPoint content if you so choose.

Basic Transitions

Transitions provide a tremendous creative opportunity for the creation of FXThemes. Presenter supports several types of transitions and they can range from very simple to highly complex. Transitions can employ both 3D animation effects techniques as well as techniques that are more like 2D digital video effects.

Note: Transitions occur between two slides but they must be associated with a single slide. Presenter handles this situation just like PowerPoint. Transitions are defined as what happens BEFORE the slide to which they are assigned.

The 3 types of Presenter Transitions and their basic definitions are:

- Clips: Clips transitions simply trigger any animation that has been designated as a "Clip" in the foreground or background scenes that are used in the FXTheme variation selected for a particular slide. Any cycling animation that the scene includes will continue playing on the Presenter unified timeline. Any Clips within the scene will start playing at the beginning of the transition. The transition will continue until the longest running Clip has completed. At that point the next slide will display.
- DVE: DVE transitions mimic the wipes of digital video effects. They employ two scenes. One contains design geometry. The second contains an animating stencil plane. In transition, the stencil plane's animation simultaneously obscures the content of the current slide while it reveals the content of the next slide. The design geometry scene (called the "Visual Representation Scene" plays simultaneously to create additional visual interest. The Stencil Scene and the Visual Representation scenes are commonly developed as a single file in 3ds max as their animation is tightly interdependent. That file is then exported twice, first omitting the stencil plane to create the Visual Representation Scene, then removing all the design geometry to create the Stencil Scene.
- Render To Texture: The most complex and flexible of the transitions techniques provided by Presenter is Render To Texture. Render To Texture transitions can also place significant demands on the end user's graphics hardware. These transitions capture sequential snapshots of specified scenes and content and map them onto specially identified geometry within the scene. This is how we achieve the effect of PowerPoint slide content being mapped onto 3D geometry that swings into view.

More information is provided on the DVE and Render to Texture transitions in the section on Advanced Techniques. For our BasicTheme we'll create just a simple Clip transition that fades the user's PowerPoint content off and back on in transitioning from slide to slide.

To get started, if you haven't already done so add one transition to BasicTheme. If you created the theme as specified in Fig 18, then you already have a transition specified. It will show up in the left column of FXDesigner as "Transition 1." If you have not added this first transition to Basic Theme, you can do that now. Click on Theme: Basic Theme at the top of the left column in FXDesigner. In the right hand window, in section C: Transitions, change the Number of Transitions to 1.

Now, click on Transition 1 in the left column. In the right hand window that appears, enter a name for the transition. It's a good idea to name transitions descriptively. This will be a simple fade out and in of PowerPoint content. A descriptive name that would tell the end users what to expect might be "Cross dissolve – Same Variation – 1 sec". Next enter the transition type. Click on the pull down list and select Clip Transition.

The default values for Foreground Clips and Background Clips are true. The effect of this feature of theme creation is to either play or suppress Clips that are included in scenes that are in foreground and background scenes of the incoming slide's selected variation. You'll recall that in our case the default variation on which we're working now includes a foreground scene which contains a Clip for the animation of our "logo." In that foreground scene, the teapot's parent object has a periodic rotation that is cycling. The teapot itself has a change in position that is referenced as a Clip. By leaving the Enable Background Clips attribute set to true, that position Clip will play as part of the transition we're creating. By separating the two controllers in the 3ds max file (rotation on parent and translation on the child), the periodic rotation of the teapot that cycles continuously can be treated independently from its bounce translation that gets triggered in during slide transition. What this means is that if a transition begins while the teapot is half way through a rotation cycle, it will not jump back to its initial orientation as it initiates the bounce clip.

Transition timing determines whether the PowerPoint content of the current slide will first transition off, and then (sequentially) the content of the new slide will transition on, or whether both transitions will occur simultaneously. For the Basic Theme we'll choose the Simultaneous option.

The final elements of this basic fade out and in transition are the visibility Clips we assign to the PowerPoint content. In Transition Out, assign a visibility Clip called visibilityOff_30. This will fade the content out over 1 second. In Transition In, assign the visibility Clip called visibilityOn_30. Note: These steps produce an FXDesigner message that asks: Enable Content Transition In (and Out) support? You first need to set TransitionOutEnable and TransitionInEnable to True. Both these visibility Clip assets were generated when we exported the file called mtlFile.max. When you are done, the dialog for your Basic Theme Content Fade Out and In transition should look like Fig. 30.

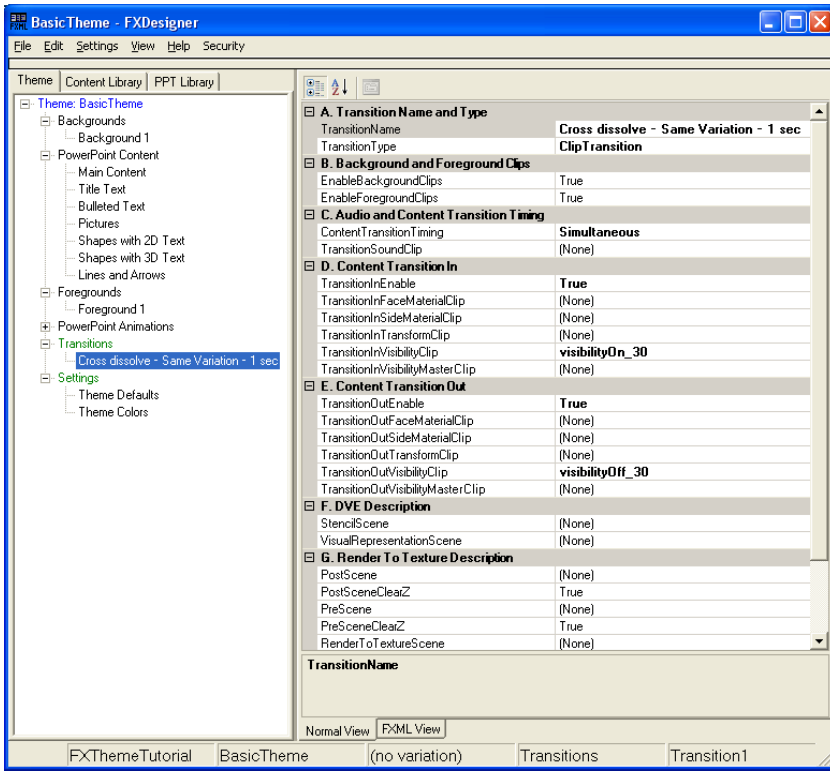


Fig. 30 FXDesigner attribute selections for creating basic Clip transition.

Now test your theme. First select File / Save All in FXDesigner. Then, re-enter PowerPoint. Once you've done that, enter Presenter with a PowerPoint file that has several slides. The content of those slides should include text, images, shapes, and lines or arrows.

The Default variation for BasicTheme should be applied to all your slides. That variation should have the concentric, slowly animating tubes as a background scene and the teapot logo scene with the transparent "Tea 4/2" company name as a foreground scene. The teapot should rotate periodically based on cycling animation.

Check to see that materials for all content types look the way you want. Verify that all selection and de-selection animations are hooked up and working.

Next, assign your Cross dissolve – Same Variation – 1 sec transition to all the slides and test it. You should see the teapot shoot up and then bounce back into position as the content from the current slide fades out and the content from the next slide fades in.

Congratulations! You've now mastered the basics of theme creation. From this exercise you can see there are numerous ways to branch out and apply your own creativity. More interesting models, more lively animation, different lighting effects, more complex and "thematic" transitions, will all add to the impact of a quality FXTheme. In addition, there are significant opportunities to add more realistic and striking materials through the inclusion of shaders. All these elements will be discussed in the following section that goes Beyond Theme Creation Basics.

Beyond Theme Creation Basics

The Beyond the Basics section expands authoring into some of the more powerful areas of FXTheme development. It builds on the concepts established in the preceding section covering the basic techniques for FXTheme authoring. In this section we will add:

- A main content scene that adds moving lights to illuminate the user's PowerPoint content
- A content slide variation to the theme
- Two types of advanced transitions (DVE and Render to Texture)
- 3D bullets
- Pixel shaders that take advantage of DirectX8.1 and DirectX9 graphics hardware

In all cases basic examples will be developed from 3ds max files that are provided. We will also learn how to package up the results of our work for distribution to other Presenter users.

Main Content Scene

A nice subtle way to add life to your presentation is with dynamic lighting. We'll do that with our main content scene so the lighting we choose illuminates our PowerPoint content. Then we'll use that same lighting scheme in the other scenes we develop.

Open the max file called cameraLight.max. Browse the scene and look at the lighting set up. A few of items to note in this scene:

- The lights have been animated by putting path constraints on circles and then moving and shaping the circles. The FXML Exporter utility, however, does not recognize path constraint percent controllers. For export, the path constraint percentage controller must be collapsed back to a position controller in order for the FXML Export utility to recognize it. Sampling can be relatively coarse. We've chosen every 5^{th} frame.
- The camera has not been moved from the default position. It is -175 units back in Y with a FOV of 50 degrees.
- The Dummy Text object is in the scene to provide a lighting test. It has its renderable property turned off so the FXML Exporter utility ignores it for export and it does not get embedded in the Main Content scene.

When you are familiar with the scene, use the FXLM Export utility to export it. For this tutorial we have chosen a file name of mainContentSceneLights. Export the scene to the Theme Folder. To complete this operation the lower (Output Options) portion of your FXML Exporter scene export dialog should look like Fig. 31.

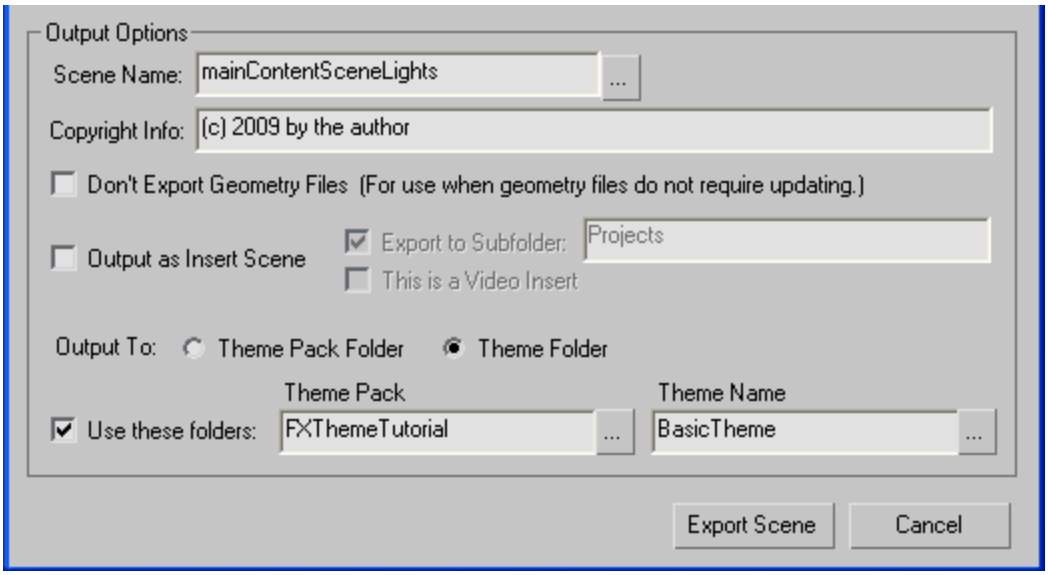


Fig. 31 – Bottom portion of FXML Export utility scene export dialog set up to generate a main content scene.

Now with the main content scene created, re-open FXDesigner. If it has been open while you were exporting the main content scene, remember to Refresh with View / Refresh Content

Tip: Note that View / Refresh Content is on a hot key (CTRL +R). As you speed up in theme creation, you'll find this a convenient short cut.

To add a main content scene to our BasicTheme, click on Main Content in the left hand column of FXDesigner (under Theme: BasicTheme / PowerPoint Content). In section B of the right hand panel select the scene you've just exported as your FXML Scene. The FXDesigner will look like **Fig**. 32.

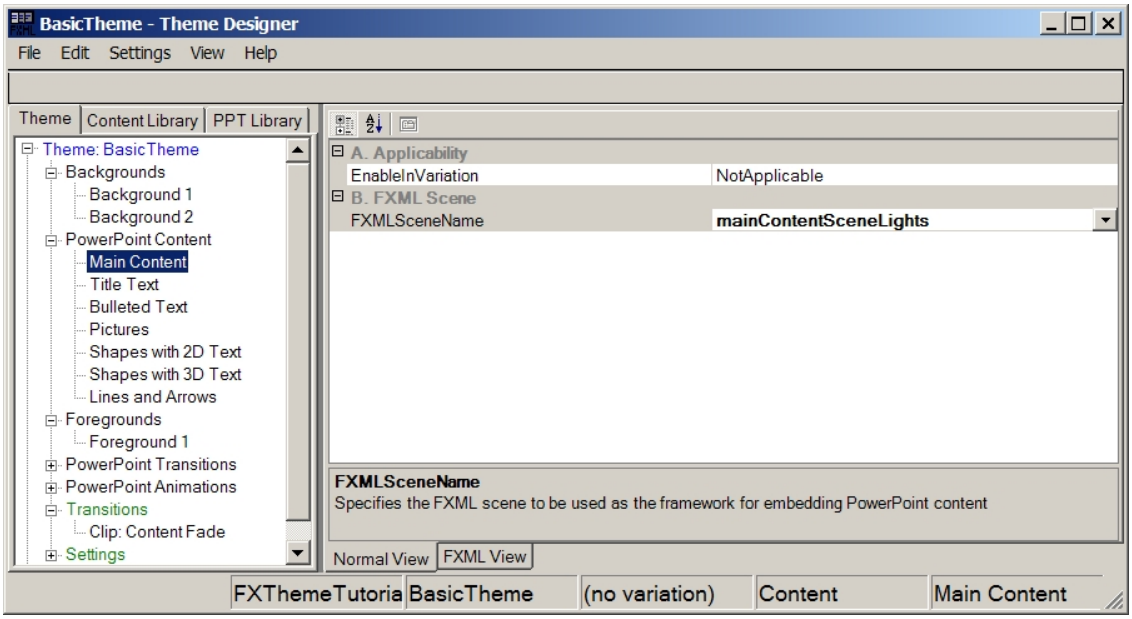


Fig. 32 FXDesigner with new main content scene added to BasicTheme.

Save the theme and go into PowerPoint. If you have been running the FXEditor, drop back to PowerPoint and re-start FXEdit. That will cause Presenter to reload the new elements of the BasicTheme. You will now see subtle lighting changes on your PowerPoint content.

In order to maintain consistency between the various scenes that make up a theme, at this point you would ordinarily need to transfer this new lighting back into the foreground scene that we exported earlier. In this case the lighting we've just exported to the main content scene has already been employed in all the 3ds max files provided.

Variations

Since only a single FXTheme can be applied to any one presentation, variations are the primary mechanism for introducing variety. That variety can be in all sorts of areas including color schemes, foregrounds, and backgrounds. It can also be aimed at providing different layouts for different types of slides. A common example of the use of variations can be seen in many of the FXThemes provided with Presenter where different variations are used to accommodate title slides and content slides.

In the BasicTheme we are creating in this tutorial, the Default variation that we have already created works well for title slides. It has a background that's subtle and moves slowly. It also has our pseudo company's logo and name represented in the corners. In our tutorial example we will now build a "Content" variation. That variation will be designed to provide enhanced contrast between the end user's PowerPoint content and the theme's background. At the same time it will allow the company brands to be continually reinforced.

To begin, open the 3ds max file called contentVariation.max. Browse the scene. Note that it is a simple path extruded "frame" around a semi-transparent plane. The path extrusion is designed to accommodate the Tea 4/2 company name and logo in the lower corner of each slide. The transparency on the plane is accomplished with a graded Targa (.tga) file. That could also be accomplished with a 32 bit PNG file. Remember that the grade information must be on the image alpha channel for it to be recognized as transparency map information by Presenter.

Also note the rotating lights. They are identical to the lighting we've just added to the main content scene. As a result, the frame and contrast plane will be lit identically to the user's PowerPoint content.

When you are finished inspecting this file, export it with the FXML Exporter utility. Use the same Theme Pack and Theme names we've referenced previously. The scene name we have used in this tutorial is "content Plane." Remember to export this scene to the Theme folder vs. the Theme Pack folder. For a snapshot of the Output Options portion of the FXML Exporter set up to do this see **Fig**. 33.

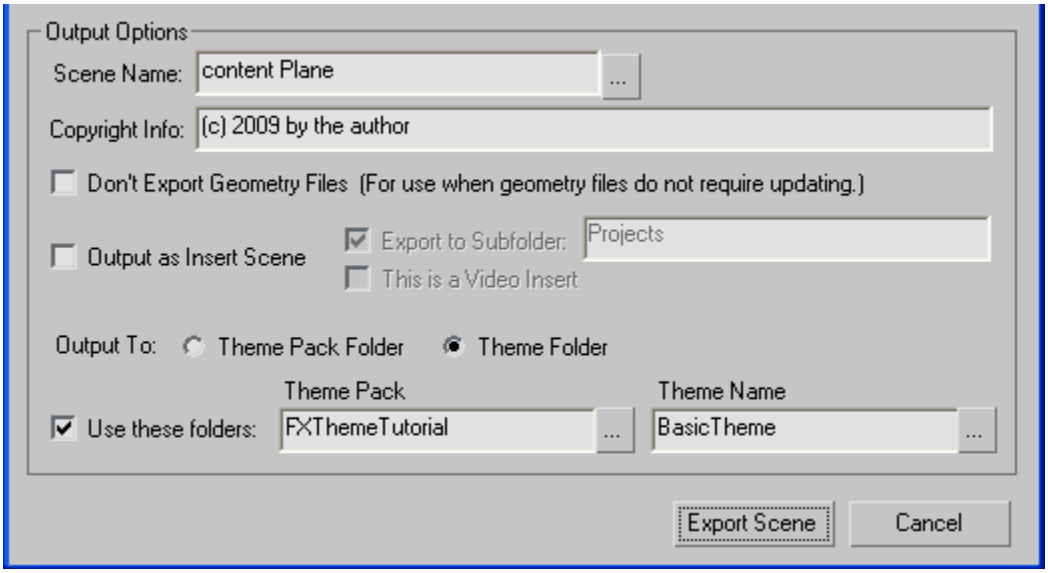


Fig. 33 – The FXML Exporter utility set up to export the content Plane scene to the BasicTheme theme folder.

To create a variation based on this new FXML scene file, we return to FXDesigner. In FXDesigner, we first need to add a variation. To accomplish this go to the File option in the FXDesigner toolbar and select New / Variation. You will be presented with the dialog in Fig. 34. Leaving number of Background and Foreground scenes at 1 each for the moment, enter the Variation name "Content"

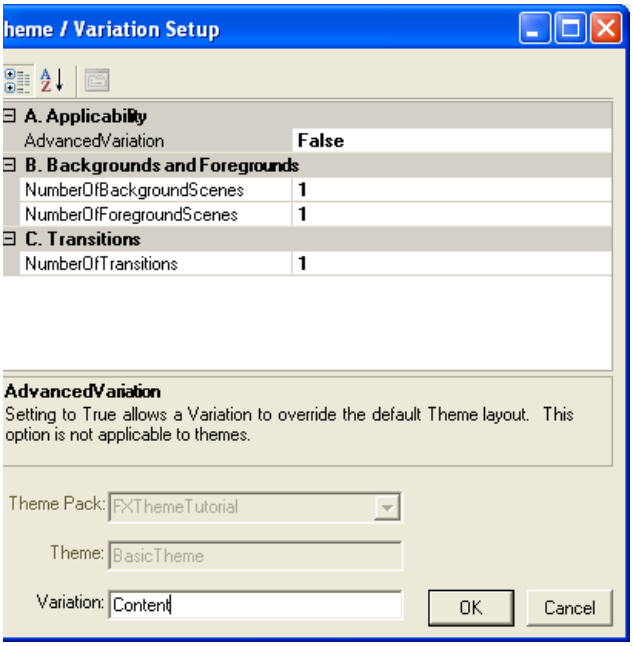


Fig. 34. The FXDesigner dialog for adding a new variation to an FXTheme. In this case a simple variation named "Content" is being added to the Theme called "BasicTheme" that is in the FXThemeTutorial Theme Pack.

In this variation, the content scene we just exported is going to be added as a new background scene. That can be done in two ways. One is to add a second background scene to the Default variation we created earlier, but leave it empty. The second is to add a second background scene to the theme that is only in this new Content variation. We will use the former approach as it allows us to create what is called a "Simple" variation. A Simple variation is where the number of foreground and background scenes is the same as in the Default variation. When variations have a different number of foreground or background scenes than the Default variation, and / or require advanced properties for those scenes, they become "advanced variations". In general it is desirable to create Simple variations whenever possible. As their name implies they are much easier to manage. They also make life easier for the end user.

To prepare the theme for creating a simple variation that uses our new content Plane scene, we click on Theme: Basic Theme and increase the number of Background Scenes from 1 to 2. When this is done, the FXDesigner UI will look like Fig. 35. Now the Default variation has two backgrounds, but the second one that we've just created does not have any FXML scene assigned. It will remain that way. It's only in the Content variation that we'll assign a second background scene.

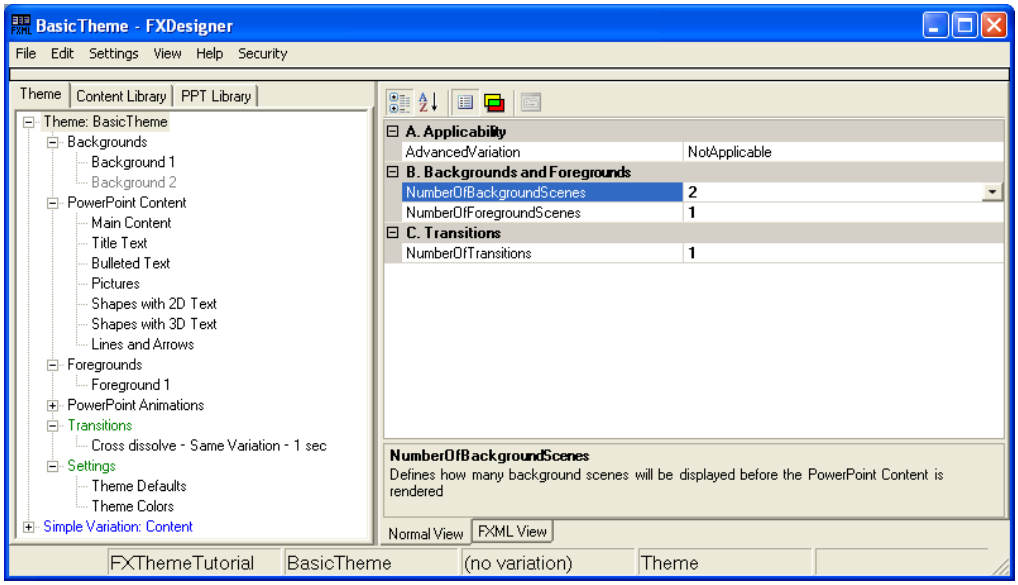


Fig. 35 FXDesigner with BasicTheme selected after changing number of Background Scenes from 1 to 2. Note that 2nd Background scene is grayed out – meaning no scene is specified.

Variations initially inherit all their attributes from the Default variation. Attribute categories that do not vary from the Default variation have their titles grayed out. Click on Simple Variation: Content and note that all the various listed attributes that make up a variation are grayed out because nothing has yet been changed. In particular, note that the listing for Background 1, Background 2 and Foreground 1 are grayed out, so currently there is no difference between the Default variation and the Content. The Content variation will include the background.fxml scene as Background 1 and the foreground.fxml scene as Foreground 1.

The first "difference" we want to create in this variation is that we want to include the content Plane.fxml scene as a second background and have it composited over the top of the Default variation's background scene. Recall that the compositing order for both Background and Foreground scenes in Presenter is that higher numbers are composited in front of lower numbers.

To add the content Plane scene, simply click on Background 2. Remember to Refresh FXDesigner (CTRL +R) first. Then in the dialog that comes up in the right hand side of the FXDesigner, in the category FXML scene, use the pull down to find and select the content Plane scene that we have just exported. You will be prompted with the question, "Enable Background support in variation?" This must be set to True. When you've added content Plane as Background 2, the FXDesigner interface will look like **Fig**. 36.

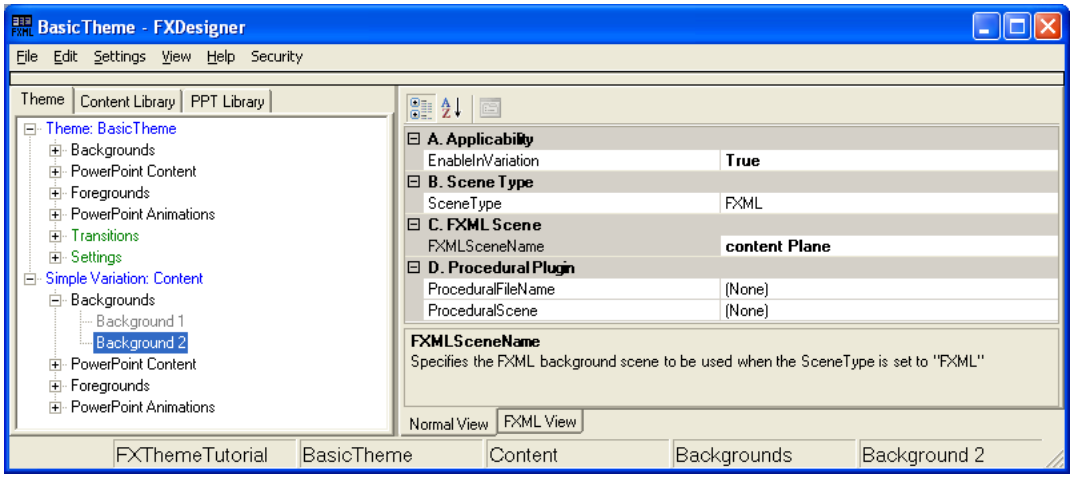


Fig: 36 – FXDesigner with new content Plane scene added as a second background to simple variation called Content.

There is one other element of the Content variation you will want to address. The content Plane provides a moderately light toned grey background for PowerPoint content. You may wish to change the materials used for bulleted text with this variation in order to provide best possible contrast and readability. When you do this, you will be prompted with question, "Enable Bullet Text support in variation?" After answering, "True", you will see that as in Fig. 36, the Bulleted Text attribute will no longer be displayed as grayed out because it now carries different instructions than the Default variation.

Once you have completed both of these changes, use the File / Save feature to save the theme. Now go back into PowerPoint. Start up the FXEditor, and note the new variation that is available for selection.

One dilemma that the new variation creates is that it adds the new frame element to slides it's assigned to. When transitioning between slides that use the Default variation and the Content variation, that frame will "pop" on or off. The solution for this jarring appearance or disappearance is the use of transitions. While all of the "Standard" transitions supplied with Presenter do a great job, they are generic. When you create custom transitions for FXThemes, they pick up on design elements of that specific theme. The more advanced custom transitions offer a wide range of creative opportunities to reinforce the design or branding of a particular FXTheme. The first of those custom effects is the DVE transition.

DVE Transitions

Digital Video Effects (DVE) transitions are generally in the class of transitions called “wipes.” They employ stencil buffer techniques that allow the content from the current slide to be wiped away and replaced with the content of the next slide. These techniques are typically 2D in nature, and in fact most of PowerPoint’s transitions are simple versions of what Presenter calls DVE transitions. The DVE transitions in Presenter, however, can be augmented with 3D elements to add visual impact and design details that are in keeping with the FXTheme.

To create a DVE transition, two scene files must be exported. The first is the stencil scene. The second is the visual representation scene. Both scenes must use the same camera as they will be composited together during the DVE transition. The most effective way to do this is create a single 3ds max file that has the elements for both the stencil and the visual representation scene combined in it and uses only a single camera. This guarantees an identical camera setup for both scenes, and makes asset management easier.

- Stencil Scene - The stencil scene contains only the geometry that will be used to wipe from the current slide to the next slide. This is literally a binary stencil effect. Wherever the stencil geometry does not appear, the current slide’s content will appear. Where the stencil exists in the scene’s camera view, the next slide’s content will appear. The position, rotation, and scale of the stencil geometry can all be animated during the transition.
- Visual Representation Scene – The visual representation scene provides the 3D design element of a Presenter DVE transition. It can be fully shaded, lit, and animated with all the details of other Presenter scenes.

Note: As in all such cases, any animation that will be triggered at the onset of a DVE transition must be designated as a Clip. In 3ds max this is done by adding the “Clip=” User Defined Property to the animating object(s). Eligible animation components are position (Clip=P), rotation (Clip=R), scale, (Clip=S), materials (Clip=M), and visibility (Clip=V). These components can be combined in the User Defined Property comment field, as in Clip=PRS.

The example 3ds max file provided with this documentation is called DVE transition.max. Open the file and review its setup. It consists of the standard camera setup with our 2 animating lights. The Tea 4/2 company logo (teapot) and a stencil plane are both parented to a dummy object that is animated over 60 frames. The dummy object has the necessary Clip=P User Defined Property comment assigned. The teapot in this example transition wipes left to right, dragging the stencil plane behind it. The stencil plane will reveal the content of the next slide.

To export the necessary two scenes for building a DVE transition from the example file, first hide the stencil plane from the 3ds max file. Then, using the FXLM Exporter utility, export the visual representation scene. Choose a scene name that will make selection in FXDesigner easy. In this example we have exported a scene file called DVE Teapot Viz Rep.

After completing the first export, return to 3ds max and unhide the stencil plane. Now select the teapot and hide that. Return to the FXML Exporter utility. This time you’re exporting the Stencil scene. In the example provided, we have named that scene file DVE Teapot Stencil. .

To create the DVE transition from the two exported scenes, you start up FXDesigner and open the FXThemeTutorial Theme Pack and BasicTheme we have used for the previous examples. Click on Basic Theme at the top of the left column. In the right column click on NumberOfTransitions and change it from 1 to 2. – See **Fig**. 37

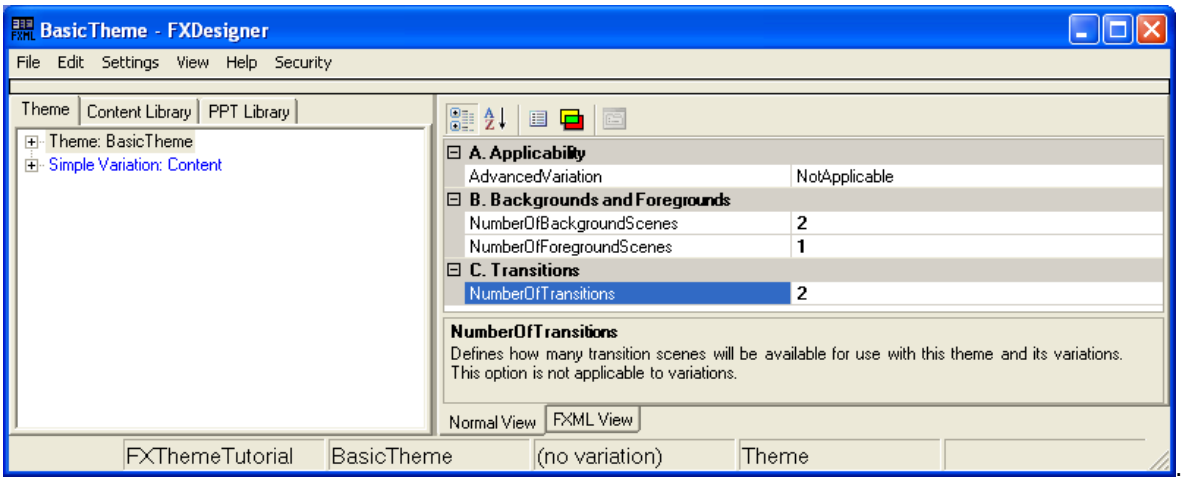


Fig. 37 FXTheme Designer UI showing modification of an existing theme by adding a new transition.

Now go to the Transitions entry in the left hand column and click on the newly added Transition 2. This will bring up a blank transition dialog in the right column. To create the DVE transition, follow these steps.

- In Section A, first rename the new transition to something indicative. In our example we've used the transition name DVE: Teapot Wipe.
- Also in Section A of the dialog, for Transition Type, use the pull down and select DVE Transition
- In Section F, labeled DVE Description, first select the Stencil Scene. Use the pulldown at the far right to show a list of available scenes. You will see the foreground and background scenes exported previously. You will also now see the two new DVE scenes we've just exported. Select the scene DVE Teapot Stencil as the Stencil Scene.
- Also in Section F, now select the Visual Representation Scene. Using the pull down at the far right, select the just exported scene called DVE Teapot Viz Rep.
- Save the FXTheme

Once complete, your transition description dialog in FXDesigner should look like **Fig**. 38.

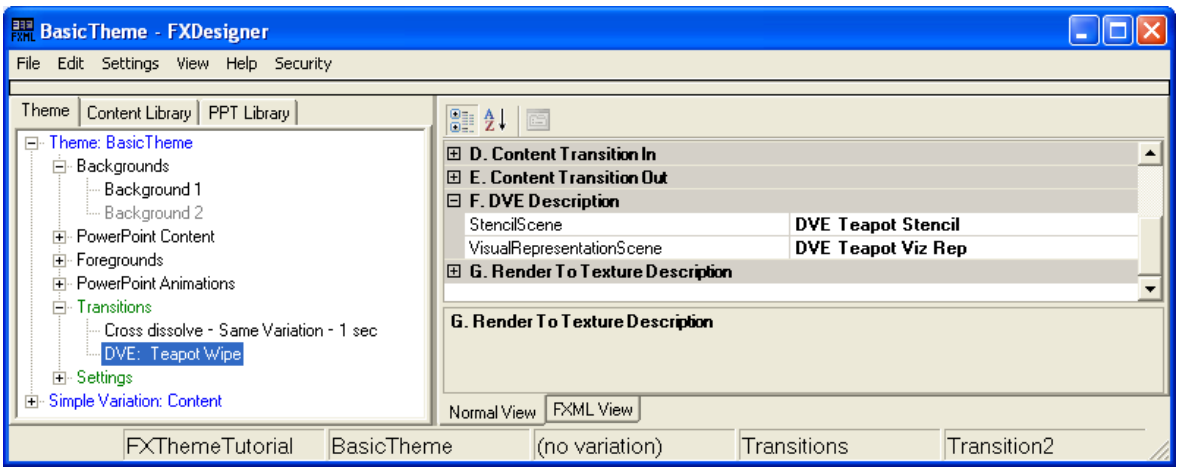


Fig. 38 FXDesigner showing the selections associated with defining a DVE transition called DVE: Teapot Wipe. Note; controls for Clip and Render to Texture transitions have been stowed away.

Now go back into PowerPoint and experiment with the DVE Teapot Wipe transition. Remember to Refresh before entering Presenter. Note how the wipe allows a smooth transition between slides that have very different content. In particular, the DVE Teapot wipe transitions nicely between the Default and the Content variations.

Render To Texture transitions

Another technique that allows for creative flexibility is the Render to Texture (R2T) transition. In this form of transition, we include the artistic option of taking real time scene snapshots and mapping them onto elements of animating geometry. Those elements can include the PowerPoint content of both the current and next slide.

To best understand what's involved in a R2T transition a good first place to look is the 3ds max file that's provided as an example. Open R2TcontentTransition.max. Note that it looks much like the contentVariation.max file. The cycling lights are the same. The geometry of the content frame looks the same, but there are significant differences. Here are specific elements of the file to review:

- The content plane object is animated. It rotates around and off the screen, then back into place.
- There are actually three contentPlane objects that are placed inside the tubular frame as children of the contentPlane Parent. These are called contentPlaneCurrentSlide, contentPlaneNextSlide, and contentPlaneTransStandin. During transition, the CurrentSlide and NextSlide planes will be texture mapped with the content of the current slide and next slide respectively.
- The CurrentSlide and NextSlide content planes will be set up to **only** display the PowerPoint that's mapped onto them during transition. There is also a semi-transparent gradient map on those planes. Because of the way we want them to only display their PowerPoint content during transition, that transparency gradient will disappear suddenly at the beginning of the transition and then pop back on once the transition is done. The contentPlaneTransStandin object is set up to take their place so this does not occur.
- All three contentPlane objects have visibility controllers on them. Halfway through the transition, when the contentPlanes have swung fully out of camera view, the visibility of contentPlaneCurrentSlide animates off, and the visibility of contentPlaneNextSlide animates on. Visibility for contentPlaneTransStandin is on for the entire transition, then shuts off once the transition is done.
- Look at the Properties / User Defined entries for all the content plane objects. The CurrentSlide and NextSlide content planes have the tags required to signal that these are render to texture meshes for this scene. Content from the current slide and next slide respectively will be mapped onto them during the transition. They also have the Clip tags required for visibility animation to be triggered by the transition. For contentPlaneCurrentSlide, those tags are:

Clip=V
R2TMaterial=CurrentSlideWithOpacity

For contentPlaneNextSlide, the tags are:

Clip=V
R2TMaterial=NextSlideWithOpacity

Note that contentPlaneTransStandin only has the Clip=V Property / User Defined entry. It does not get mapped with any PowerPoint content, but its visibility animation is triggered during transition by the Clip=V entry. Note also that it is placed just slightly further from the camera than the Current and NextSlide contentPlane objects. This assures that it renders behind the PowerPoint content that will be mapped to those contentPlanes during transition.

- The object that actually rotates the three contentPlane objects in and out of frame is the dummy object that is their parent. It is called contentPlane Parent. Note its Clip=R tag in Properties / User Defined. That tag causes the rotation of the parent and its children to be triggered at the start of the R2T transition.
- There are also two plane objects that are designed to transition the slide titles. They are named TitlePlaneCurrent and TitlePlaneNext. They have the same visibility animation as the content planes where TitlePlaneCurrent's visibility animates off, and TitlePlaneNext's visibility animates on, midway through the transition. They also have Clip tags that are similar to the contentPlane objects. The only difference is that scale is included as part of the transform Clip. For TitlePlaneCurrent, those Properties / User Defined tags are:

Clip=SV
R2TMaterial=CurrentSlideWithOpacity

For TitlePlaneNext, the tags are:

Clip=SV

R2TMaterial=NextSlideWithOpacity

- Mesh names become an important consideration with this scene as it is very similar to the Content Variation scene. The new content plate mesh names assure that there are no conflicts with that mesh in the Content scene, but the frame is the same in both. That doesn't matter as that mesh is identical in both scenes so overwriting it by exporting this scene does not cause problems with the Content Variation. If the frame mesh were different at all between the two scenes it would require a different name
- One of the more delicate parts of doing an R2T transition is getting the texture mapping perfect on the meshes that will get the current and next slide content. If it's not just right there will be a perceptible shift in the position of content either at the beginning of the transition or at the end. Achieving a precise match will likely take multiple iterations. See Fig. 39.

Tip: Use the single step time tools in Presenter to aid close inspection of mapping results. Use Preferences / Advanced Options to turn on Developer Keyboard Controls. F11 enters transition and begins single step mode. F10 steps forward one frame (1/30th of a second). F12 steps backwards 1/30th of a second. F9 toggles into and out of single-step mode.

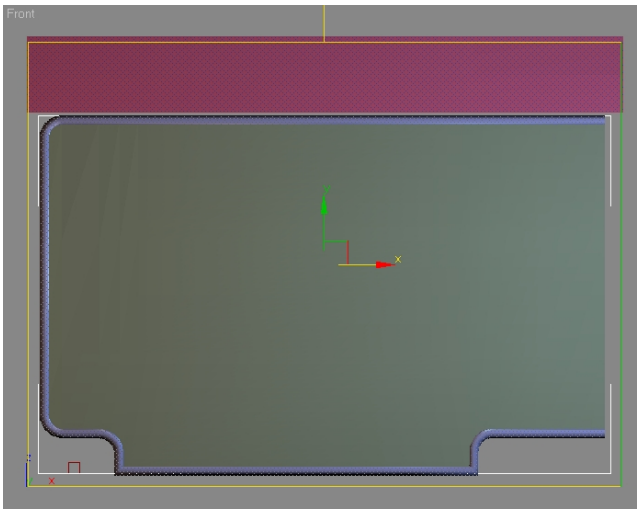


Fig. 39 shows the Front camera view of the R2TcontentTransition 3ds max file with the position and size of the planar UVW Mapping modifier Gizmo displayed. Gizmo extent is the thin yellow line at top, left, and bottom, and green on the right.

Once you are familiar with the R2TcontentTransition scene, use the FXML Exporter utility and export it to the BasicTheme theme folder. For this tutorial, we have used the file name R2T Content to Content. This "Content to Content" designation is used to indicate that the transition is designed to transition between slides that both use the Content variation.

Next we'll make a special foreground scene to use as part of the R2T transition. The foreground scene that is used in the Default variation has the teapot that bounces up and down in transition. That would have it crossing the path of the content planes as they rotate in and out of frame during transition. Let's make a different foreground scene that works better with the R2T transition.

Open the 3ds max file logoForegroundTeapotPour.max. You'll see that it is an exact copy of logoForeground.max except that the Clip motion on the teapot itself is a rotation instead of a bounce. As a result, the Properties / User Defined tag for the teapot is Clip=R. This rotation triggers immediately at the start of a transition. It is intended to mirror the rotation that we just created on the content frame in the R2T transition. Once you're familiar with the workings of the logoForegroundTeapotPour.max file, bring up the FXML Exporter utility and export it to the BasicTheme theme folder. For this tutorial we have chosen the name foregroundTeapotPour for this FXML file.

We have now created the scenes necessary for our R2T transition. The final step is to use the FXDesigner to assemble those scenes and build the transition. Go back into FXDesigner. If it has been running while you were exporting the new scene files, remember to click on View / Refresh Content (CTRL + R) so that it recognizes them.

As with the addition of the DVE transition, your first step is to click on Theme: BasicTheme at the top of the left column and change the Number of Transitions in the right hand dialog from 2 to 3. You will see Transition 3 added to the Transitions list in the left column. Click on Transition 3 to open up the right hand dialog with all the design controls for that transition.

First name the transition something that is descriptive of what it is and how it should be used. As we've observed, this transition is designed to operate between content slides. For this tutorial we've chosen the transition name - R2T: Content to Content. Then for Transition Type, pull down and choose RenderToTextureTransition.

Now we'll select the scenes for the various components of the transition. The Render to Texture transition supports a single background and single foreground scene. They are called the PreScene and PostScene respectively. The PreScene is laid down first. The Render to Texture scene is composited over that. Then the PostScene is composited over the top of both the previous two scenes.

For our Content to Content R2T transition, first choose the background scene as the PreScene. Next choose the R2T Content to Content scene as the Render to Texture Scene and our new foregroundTeapotPour scene as the PostScene. Finally, turn on the True setting for both the UseOnlyContent attributes (CurrentSlide and NextSlide). When you've made those selections your FXDesigner dialog will look like **Fig**. 40.

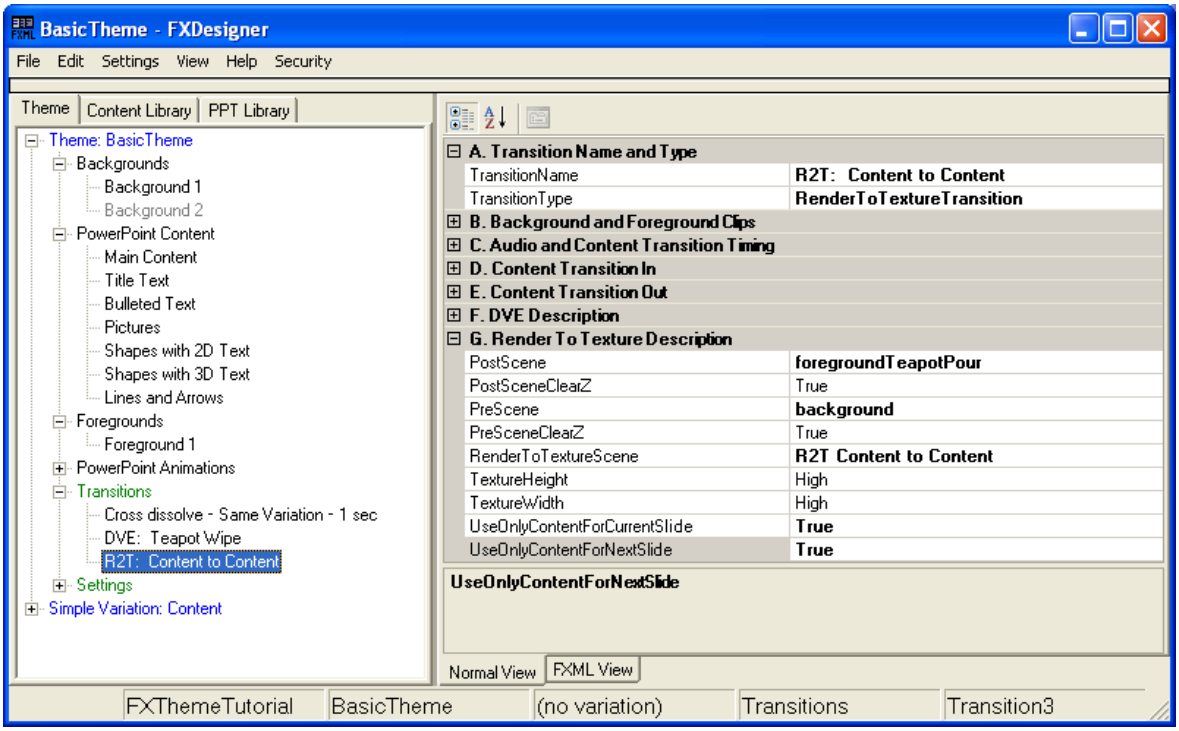


Fig. 40. FXDesigner showing setting for new Render to Texture transition called R2T: Content to Content. Note: Clip and DVE transition controls have been stowed away.

Save the theme and go into Presenter (remember to Refresh first). You'll now see the R2T: Content to Content listed as an available transition. Experiment with the use of this transition. You'll note that it is intended to work as a transition between slides that both use the Content variation. Note the pop that occurs when one of the slides uses the Default Variation. The Content plate appears or disappears suddenly ("pops") as the R2T scenes are swapped in.at the beginning of the transition and out when it's done.

Render to Texture transitions provide several additional options. Those can be seen in FXDesigner. Those options are:

- PostSceneClearZ: The default value is True. When True, the Z buffer is cleared after drawing the PostScene. If this value is set to False the Z buffer is not cleared after drawing the PostScene. Its contents can then interact with the Render To Texture Scene. In the latter case it is the near and far clipping plane settings of the two scenes which determine if content within them will interact.
- PreSceneClearZ: Same as PostScene. Default is True. If set to False, the content in the PreScene can interact with the content in the Render to Texture Scene.

- Texture Height and Width: These are default resolutions for texture creation. Render to texture transitions can use large amounts of texture memory – especially if the R2T meshes takes up a large part of the screen and the user is presenting on a high resolution system. The Texture Height and Width settings allow you to control the resolution at which the R2T transition will run. Those settings can either be relative (High, Medium, Low) or can be specific numbers that are powers of 2. In general leave them at High.

- UseOnlyContent (Current / Next): This controls whether just the PowerPoint content or the PowerPoint content plus the theme's other design elements (foregrounds, backgrounds) are mapped onto the R2Tmesh. Default is False so both the PowerPoint content and all theme background and foreground elements that are overlaid by the R2T mesh will appear on those meshes as they animate through the R2T transition.

  To gain an understanding of the UseOnlyContent option, experiment by turning it to False in the above tutorial R2T transition. You'll see that with the option turned off the contentPlane and titlePlane objects get images of both the current and next slide's PowerPoint content, but also have an image of the background scene mapped onto them. Because they animate, this does not look right. It's as if a chunk of the background scene gets torn off and moves with the PowerPoint content. By turning on the UseContentOnly option and coupling that with the R2TMaterial Property / User Defined entries that append "WithOpacity", we see only the PowerPoint content mapped on those meshes as they move and scale during transition.

  Another useful experiment is to eliminate the WithOpacity from the Property / User Defined entries and then re-export. With UseOnlyContent set to True, instead of seeing through the R2T contentPlanes to the background, you will see solid black behind the PowerPoint content that's mapped to those planes.

Defining Custom 3D Bullets

A nice additional touch that can be added to an FXTheme is 3D bullets. Theme bullets are displayed with the PowerPoint content so they are viewed by the theme's main Content camera. A bullet's geometry can take the form of any single .X file you choose from your Theme or Theme Pack (or even their Media folders). Because the .X file contains no information materials, you assign a material to the bullet in the Theme Designer. You do not need to worry about cameras or lighting because the bullet is viewed within the Main Content scene along with the rest of the PowerPoint content. There is no explicit "export bullet" option in the FXML Exporter utility, as any exported .x file is a valid bullet choice.

Tip: If you need to export models for bullets, export them to an Insert Scene and move the resulting .X file(s) back to the theme folder. This will keep your theme folder from having erroneous .fxml scene descriptions.

Bullet geometry is automatically sized relative to the user's PowerPoint text and FXDesigner also provides a scale control, so you do not need to be concerned about object size when creating 3D bullets. However, you do need to be mindful of its pivot point. The object will be placed with its pivot point at the bullet text character's origin. For objects coming from 3ds max, bullet geometry should be built to have zero rotation values in all coordinate channels (X,Y, Z) when oriented as desired and viewed through the default Presenter camera.

An example file called 3D Bullet.max is provided to illustrate creation of bullet geometry. In the case of primitive objects like the teapot or a cone, proper orientation can be achieved by constructing the object in the Top view. It then has zero rotation when viewed as desired in the default Presenter camera which is similar in its orientation to the 3ds max Front orthographic view. It looks at the origin from a position of (0, -175, 0). Bullet object location (in world space) does not affect the eventual bullet placement in Presenter. Only the bullet object's pivot location does. A bullet's geometry can be made to move away or towards its text by the movement of its pivot point.

Export the example file using the FXML Exporter utility. You'll see that it is set to export to an Insert Folder instead of to the BasicTheme folder. This is to avoid creation of FXML files that will not be used in the theme folder. The only element that you need from this export is the .x file that is created. That is called Teapot Bullet.x. It will be found in the newly created Teapot Bullet Insert folder. Copy the .x file and move it manually to the BasicTheme folder.

Now we'll create animation for the bullet. Custom bullets are affected by bulleted text animation, but they can also have their own cycling animation. That animation is subject to the same orientation requirements as the clip based animation that controls the animation of PowerPoint content. Therefore, it is most straightforward to create the bullet's animation in the same file as you use to create motion assets for your theme. The geometry that you use to create that animation should be made a child of the same X rotated object that is parent to all your motion file animation clips.

In the motionsFile.max file that is supplied with this tutorial, two objects that are set up for bullet animation are included. Both are represented geometrically as teapots. They are called:

- rotateY_120_noClip, and
- rock_120_noClip

Both objects are children of the grid object called prodObjParent that is rotated 90 degrees in X. Both objects are then animated so that they move appropriately when viewed by the default Presenter camera. The only difference between these and the other motion assets in the file is that they do not have Clip tags in their Properties / User Defined fields. Instead they have out of range controllers set to cycle their animation continuously. This motion file has already been exported so you will not need to repeat that. The bullet animation asset is already available.

The bullet's material is assigned in the Theme Designer because .x files do not contain material assignments or definitions. Because a bullet can only reference a single .X file (rather than a scene) only one material can be assigned to a bullet. Multiple colors can be achieved by assigning a material with a diffuse map.

With the assets prepared, we can now add the 3D bullet to our FXTheme. Bring up FXDesigner. If it has remained up during your export of the bullet file, remember to Refresh Content so that the new asset is recognized. In your Default variation, click on Bulleted Text in the left column. Scroll down in the right column to section H.

The H section in the FXDesigner's PowerPoint Content / Bulleted Text panel allows you to define custom 3D bullets that can optionally replace the font bullet coming from PowerPoint. Defining just Level 1 will replace all bullet levels with this one definition. Defining both Level 1 and 2 will replace all levels below Level 2 with those of Level 2, etc.

In FXDesigner, now choose Bullet Geometry. This prompts "Enable Bullet Support in variation?" After answering, "True", select the Teapot Bullet.x file. Next choose a material. You will see that all the non-animated materials that you have created as assets for this theme are available in the FXDesigner list that's provided. Initially stay with a scale of 1. You can adjust that to your taste subsequently. Finally, choose an animation. The two that you will find available are the two listed above that have come from the motions file. Both are non Clip-based animations.

You can optionally define different animations for level 2 bullets and so on. If you do not, the lower level bullets will simply inherit the characteristics of the last level above them that is defined. A finished FXDesigner UI for custom bullet definition will look like **Fig**. 41.

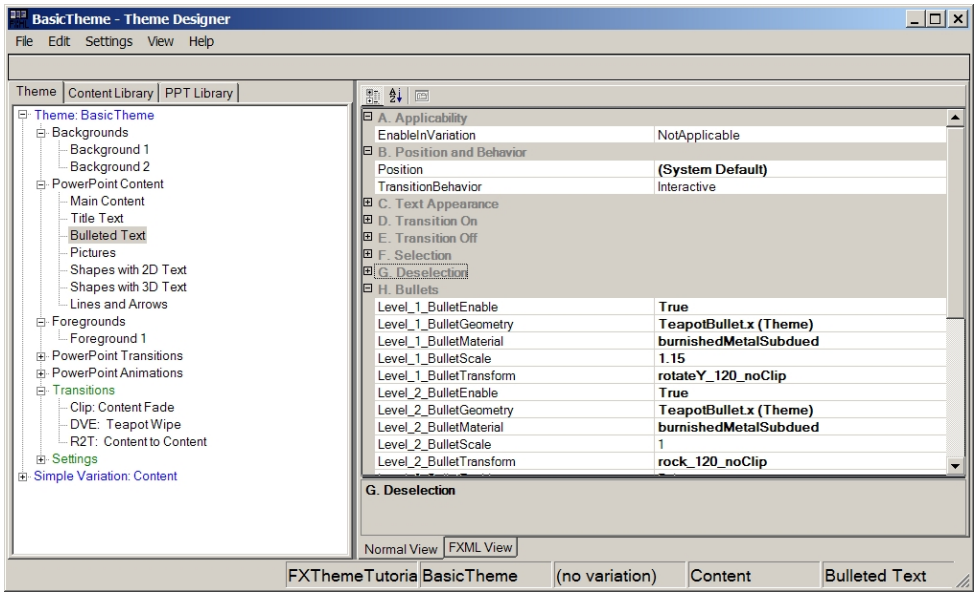


Fig 41. FXDesigner showing definition of custom bullets for the BasicTheme

Note: If you have made any changes to the Bulleted Text definition for the Content variation, you will also need to redefine custom bullets in that variation.

Now save the theme and go back to PowerPoint to view the result (remember to Refresh themes if you have left PowerPoint open). In order to see the custom bullets you've created when in Presenter, the option to Use PowerPoint Bullets must be turned off in the Presenter Preference dialog.

Applying Shaders

Presenter is a fully DirectX9 capable application. It allows the inclusion of custom DirectX8.1 and DirectX9 shaders within FXThemes and Insert Scenes. In this section we will describe the process of adding a custom shader to elements of our BasicTheme development. Example files are provided. Prior knowledge of the concepts involved in custom shaders is required. More information on creating custom hardware shaders can be found on the web in game developer forums and in information posted by both ATI and NVIDIA.

General Info

In order to add custom DX8 or DX9 shaders to Presenter content, it is necessary to either create a custom plug-in for Presenter, or use one of the shader plugins that come with the product. We will not discuss the details of writing a custom shader plugin in this document.

Presenter includes two general purpose shader plugins, both of which are based on the Microsoft DirectX .fx file format. The two plugins differ only in the semantic connection that describes how data is communicated between the shader .fx file and Presenter. These shader plugins are called StdFXShader and StdRTZenShader. Source code for both shaders is located in the OfficeFX\SDK\ProcObjects folder. Both are compiled into the ProcObjects.dll file in OfficeFX\Repository\Plugins.

The difference between these two shader plugins involves details regarding the construction of the shader code. Once the shader has been implemented, the process of using it for export from 3ds max is the same in either case. In this tutorial, we will focus on that latter process. The above referenced shader plugins will not be modified in any way.

When creating a shader for use in Presenter, the shader-writer must create a DX9-compatible .fx file along with an accompanying XML file that tells Presenter how to load the .fx file and hook up any user-settable parameters. Several example .fx and XML files are provided in the standard Presenter install. Those are located in the OfficeFX\Repository\Media\Shaders folder. (Note that any valid High-Level Shading Language syntax can be used in the .fx file, but that any Cg or .CgFX syntax – as defined by nVidia but not Microsoft – will not work.)

Custom Shader Basics

In this document we will concentrate on how to use these existing shader files and create references to them in 3dsmax such that they appear in FXThemes and Insert Scenes. We'll then cover ways in which elements of the 3dsmax files can be used to control attributes of the shader.

The files that we will use and which will allow you to experiment with and learn the process of custom shader inclusion and control are as follows:

- shaderInsertScene.max: A 3ds max file with proper setting for creating references to a custom shader on our teapot insert scene. This file is located in the AuthoringToolkit\FXThemeMaxFiles folder.
- NewBumpReflect.xml: A shader mapping file that gets placed into the FXML file output from 3ds max by the FXML Exporter utility. The attributes of NewBumpReflect.xml override the material properties of specified elements of the 3ds max scene with attributes of the specified shader. This file is a slight modification to a standard Presenter inclusion file that is located in the OfficeFX\Repository\Media\Shaders folder. This file is included in the AuthoringToolkit.zip file and is extracted to the AuthoringToolkit\FXThemeMaxFiles\Shaders folder.
- BumpReflect.fx: A High-Level Shading Language shader description file that defines the look of the BumpReflect shader. This file is a standard Presenter inclusion that is located in the OfficeFX\Repository\Media\Shaders folder. BumpReflect is a DX8 shader.
- Bump_OFXlogo2_mmap.dds: A high resolution mip-mapped image file that will be used as a normal map within the BumpReflect shader. The file is included in the standard OfficeFX\Repository\Media folder. (Note: This one is NOT in the shader folder. It's one level up).
- lightcubestar.dds: A high resolution mip-mapped image file that will be used as a cube map within the BumpReflect shader. The file is a standard Presenter inclusion that is located in the OfficeFX\Repository\Media\Shaders folder.

A simplified description of the process is as follows:

1) Place the shader.xml file that you'll be working with either in the same folder as your 3ds max files or in the 3ds max matlibs folder.

2) In 3ds max create an object and assign a material to it. This material is called the "base material."

3) Enter a reference to a shader XML file in the user defined properties for that object.

4) Export using the FXML Exporter utility. This shader.xml file content that you've put into the 3ds max matlibs folder will be included as part of the exported FXML scene description. It will be associated with the object you have tagged with the user defined property "shader include" reference.

5) If you are building an FXTheme that incorporates custom shaders, use FXDesigner to include the scene you've just exported as part of that FXTheme. If building an Insert Scene, you're done.

6) Run Presenter and use the FXTheme or Insert Scene you've created.

During the FXML Export process, the FXML Exporter will embed the content of the specified shader.xml file in the scene.fxml file that is created from the 3ds max scene. Those shader instructions will be associated with the mesh object for which the user defined property shader reference has been entered. Inside Presenter, the exported object will attempt to load the associated shader.fx file and look for the first shading technique that works with the runtime graphics hardware. If such a shading technique is found, it is used instead of the material that was assigned to the object (in the normal fashion) within 3ds max. If no valid shading technique is found, the base material assigned to the object in 3ds max is used in place of the .fx shader.

By way of example, suppose a shader is written that requires DX9 hardware to work. At runtime, if Presenter is running on a full DX9 machine, that shader will be loaded and used. On the other hand, when Presenter is running on DX8 hardware, the custom shader will fail to load and Presenter will drop back to using the base material assigned to the object in max.

Mechanics and Tutorial Specifics

Now let's go through the process using the specific files provided and referenced. Steps are:

1) Place the provided file called NewBumpReflect.xml into either the Program Files\3ds max\matlibs folder, or into the folder that you're using to hold these tutorial 3ds max files. NewBumpReflect.xml is in the FXThemeMaxFiles\Shaders folder.

2) Open the file shaderInsertScene.max in 3ds max. You'll note that in most ways it is identical to the previous insertScene.max file we used to create our Insert Scene. The key difference is in the user defined shader reference. Select the teapot and display its Properties / User Defined reference. You will see:

"ShaderIncludeFile=NewBumpReflect.xml", or alternatively the shortcut syntax of

"SIF=NewBumpReflect.xml"

3) When you've finished inspecting the file, bring up the FXML Exporter utility and export it as an Insert Scene called "shader Teapot."

4) Open the file NewBumpReflect.xml and review its syntax. You will observe that it refers to the StdFXShader that has been compiled into the ProcObjects.dll. It calls the BumpReflect.fx file that is in the OfficeFX\Repository\Media\Shaders folder. It also calls bump_OFXlogo2_mmap.dds as a normal map and lightcubestar.dds as a cube map. Note that their paths are non-specific. This will allow the standard Presenter search process to locate them as they are in the OfficeFX\Repository\Media folder and Shader sub-folder.

5) Now open PowerPoint and then Presenter. You will find the new Insert Scene called shader Teapot is available to be added to any slide. BumpReflect is a DX8 shader. It can be displayed on both DX8.1 and DX9 capable hardware. You should see a deep blue reflection map with a detailed bump map that incorporates the Instant Effects logo.

Caveats

Because the shader assignment is made through a user defined property instead of by directly modifying a material within 3ds max, it is important to make sure that you create a one-to-one match between materials and shaders. More precisely, if a 3ds max material, called Material A, is augmented by an SIF user property on one object, then the same shader user property should appear for any other objects using Material A. This is because under the covers, the XML shader specification is spliced into the exported definition of Material A. So, if some objects with Material A have an SIF user property value and others don't, the exported definition for Material A would not be well-defined.

Understanding the Shader XML File

Here's an example of a shader XML file called BumpPlastic.xml. It too is a standard Presenter inclusion. BumpPlastic.xml is a DirectX 9 shader. It exhibits more control elements than BumpReflect.xml, but it will not show on DirectX8 hardware.

```
<ProceduralShader>
    <AssemblyName>ProcObjects.dll</AssemblyName>
    <ClassName>ProceduralObjects.StdFXShader</ClassName>
    <Description>
      <FXFile>bumpPlastic.fx</FXFile>
      <TransposeMatrices>true</TransposeMatrices>
     <IsTransparent>false</IsTransparent>
      <Parameters>
          <Parameter>
           <Name>lightPos</Name>
```

```
      <Value>-100 -100 100 0</Value>
    </Parameter>
    <Parameter>
      <Name>lightColor</Name>
      <Value>1 1 1 1</Value>
    </Parameter>
    <Parameter>
      <Name>ambiColor</Name>
      <Value>0.07 0.07 0.07 1</Value>
    </Parameter>
    <Parameter>
      <Name>surfColor</Name>
      <Value>1 1 1 1</Value>
    </Parameter>
    <Parameter>
      <Name>specExpon</Name>
      <Value>12</Value>
    </Parameter>
    <Parameter>
      <Name>bumpy</Name>
      <Value>1</Value>
    </Parameter>
    <Parameter>
      <Name>colorTexture</Name>
      <Value>shaders\default_color.dds</Value>
    </Parameter>
    <Parameter>
      <Name>normalTexture</Name>
      <Value>shaders\default_bump_normal.dds</Value>
    </Parameter>
  </Parameters>
 </Description>
</ProceduralShader>
```

The AssemblyName and ClassName fields tell Presenter where to find the (built-in) plugin for supporting this particular shader.

Inside the Description block, the FXFile field specifies the name the Microsoft DirectX .fx file to load when using this shader. The TransposeMatrices and IsTransparent fields are technical and won't be discussed here.

The Parameters block contains descriptions for any shading parameters that can be controlled outside of the actual .fx file itself. Each individual Parameter block contains a name-value pair. The Name field refers to the name of a variable ("uniform parameter") defined in the associated shader .fx file, with the Value field being the new value to assign to that variable.

To reiterate, Presenter uses a standard search mechanism when attempting to load image files. The same search mechanism applies to shader texture files, such as those specified by the colorTexture and normalTexture value fields above. Alternatively, you can put in a fully-qualified hard-coded path, but then, of course, the shader XML file would not be portable.

Modifying the XML File

There are two ways to modify a shader XML file for use with the 3ds max authoring process. The simplest is to use a text editor to edit the file and replace existing values with new ones of your choosing.

Note that if you don't want to overwrite the original values found in a shader XML file, you can save your modified XML file to a different name and then use the new file name in your SIF user property specification as we have done with NewBumpReflect.xml.

Also note that if you don't want to specify the value for a particular parameter, but rather want to make sure that you are using the shader's default value, you can delete the entire Parameter section for that particular parameter.

An alternative way to define parameters is to tell the shader to use values from variables found in the base 3ds max material.

For example, you may wish to drive the surfColor variable in the example above by the diffuse color of the object's base material. For this, you need to change the Value field in the Parameter definition as follows:

Original

```
<Parameter>
 <Name>surfColor</Name>
 <Value>1 1 1 1</Value>
</Parameter>
```

Modified

```
<Parameter>
 <Name>surfColor</Name>
 <Value>[Diffuse]</Value>
</Parameter>
```

The "[Diffuse]" syntax (with mandatory square brackets) tells the shader plugin to send the diffuse color value of the base material to the .fx shader at runtime. This works even if the diffuse color of the base material is animating – thereby giving you a simple way to animate shader parameters.

Note: The mapping is logical, as opposed to literal. By this we mean that the material diffuse color can be used to drive any shader parameter that takes a vector as a value. So, the diffuse color could be used to drive the lightColor or ambiColor (or even lightPos) variables in the above example.

Supported Logical Parameters

The following paragraphs describe which base material values can be used to drive shader values, and what, if any, constraints apply to those values:

- Anywhere a reference is needed to a color, you can use "[Ambient]", "[Diffuse]", or "[Specular]." If these colors are animating in the base material, then the values passed to the shader will also be animating. Also note that the Diffuse value in FXML materials is often white (1 1 1) even if it was non-white in MAX, since applying a diffuse texture causes a material to have white plugged into its diffuse channel in the exported material. (That's just a head's up so you don't waste time using what you think are animating diffuse values along with a diffuse texture in max.)
- Anywhere a reference to a bitmap file is expected in a particular shader's parameter list you can use "[DiffuseMap]", "[ReflectionMap]", or "[OpacityMap]". This will cause that shader to receive the name of the bitmap associated with the containing material's like-named-texture.
- Anywhere a reference to a scalar value is needed you can use "[DiffuseTexAmount]", "[OpacityTexAmount]", "[ReflectionTexAmount]", "[Opacity]", "[SpecularLevel]", "[SpecularSoften]", or "[Emmisive]". Again, animating values carry over to the shader.

As mentioned above, all of these mappings are logical, not literal. This means, for example, that you can use an something like the Opacity value from the material to drive any scalar parameter in the shader, or even use the Ambient color to drive Diffuse in the shader.

Implementation Notes and Trouble-Shooting

In this section we cover some topics that might lead to confusion in a real world scenario.

1. If you reference [DiffuseMap], for example, and forget to assign the Diffuse Map in the base material, or give it a map Presenter can't find or Direct3D doesn't like, the exported scene probably won't load.
   Actually, whether a missing value is acceptable or not is entirely up to the .fx shader. If it is written to work whether or not a particular map is defined or is valid, the scene will load fine under any circumstances. If the .fx shader is written to absolutely need a particular map that isn't defined, the shader load, and hence the scene load, will fail. The outcome for a particular shader is entirely dependent on how the shader is written.
   Just for the record, most .fx shaders don't do any "error detection or correction", so missing values will, in general, cause a load failure. This is because shaders are (quite appropriately) written for runtime efficiency, not input-parameter robustness.

2. When doing mappings, you really need to look at the corresponding .fx file so you know what the various Parameters are referring to (e.g., "ambient" may mean ambient surface color in one shader and ambient light level in another). Also take a close look at the parameter's valid ranges. For example, it is not uncommon for values to be 0.0 to 1.0 in the shader, while the scalars coming from the Max material are typically 1 to 100. In this case you will see a distinct change from 0 to 1 but nothing more from 2 to 99. To correct such associations, you will need to modify the .fx code appropriately (e.g., by immediately dividing the variable by 100).

3. When specifying names for bitmaps, getting rid of the "shaders\" path found in the examples will result in Presenter looking in all the standard places, rather than having to have the map in a \shaders subfolder.
   Also note that you can pre-pend any other name (other than "shaders") to separate out special maps into special subfolders of the media directories, should you have a need to do that. (The Presenter search mechanism just appends the supplied file name to the various media paths, so by specifying something like "CubicMaps\MyCubicMap1.dds," Presenter will look in the various Media\CubicMaps folders for MyCubicMap1.dds. In this fashion, you can organize all your cubic reflection maps in their own subfolder off your preferred Media folder. This is especially useful for non-color maps such as normal maps, noise maps, and volumetric function approximation maps.

4. Be very careful with DDS maps, as each is a "flavor" and many are not compatible - some are cubics, others normal, others are mip-maps, etc. Basically do an export and test in Presenter as soon as you assign one to see if it loads, because if it doesn't, you know it's wrong. Same goes for using 2D bitmaps like jpg's instead of DDS files - many times you can, sometimes you can't - you actually have to look into the .fx file to see how it's being used to be sure.

5. The FXML Exporter looks in your 3ds max5\matlibs folder for a shader .xml file if it doesn't find it in the same directory as the max file. Thus, you can safely move all your .xml files from \shaders to \3dsmax5\matlibs and then you can edit max files from any location without worrying about having .xml files there too. Or you can place a scene-specific set of .xml files in the scene folder so that you don't have to worry about name clashes with other projects.

6. If you modify base material parameters for the sole purpose of controlling unrelated shader parameters, you may get undesired results when/if the shader fails to load. For example, if a shader fails to load at runtime due to insufficient GPU power, your modified base material will be used "as is" – thereby using the modified parameters in a literal fashion rather than through the logical connection provided by the shader.

FXPackager

Once you've created a new FXTheme or Insert Scene, you will want to package it up and send it to other Presenter users. The FXPackager utility gathers all the assets and scenes from your Theme Pack and Theme folders, or from your Insert Scene folder, and puts them into a specially formatted Presenter Asset file. That file is unpackaged for use in Presenter from the Product / Install FXAsset utility on Presenter toolbar in PowerPoint.

To use the FXPackager, it must first be installed in the OfficeFX folder as described in the section titled, **Installing FXDesigner and FXPackager.** With your BasicTheme completed, bring up the FXPackager, either by double clicking on the FXPackager.exe file in the OfficeFX folder, or by creating a shortcut, putting that somewhere more convenient, and double clicking on that. When FXPackager comes up, if it is not already selected, choose the FXTheme tab. In the first pull down below the FXThemes tab, select the Theme Pack. In our example it will be the Theme Pack we've created called FXThemeTutorial. You will then see a list of all the FXThemes in that Theme Pack. In this tutorial we have created only a single FXTheme which we called BasicTheme. When you've made those selections, your FXPackager UI will look like Fig. 42. A radio button option allows you to select individual FXThemes from the Theme Pack or to package up the entire Theme Pack. A check box below the FXThemes list allows you specify inclusion of asset folders such as Media or Shaders that are in the Theme Pack folder, and contain assets that may be shared by multiple themes in the pack.

New in FXPackager with the release of Presenter is the ability to include specified Insert Scenes, Video Insert Scenes, and even separate Transition packs in an Presenter Asset file. This allows you to create and deliver more interesting and complete Presenter Asset sets in a single file. It simplifies the installation process for your colleagues and customers.

To include Insert Scenes in a FXPackage file with your FXTheme, click on the Insert Scene tab and select one or more Insert Scenes. The FXPackage file included with this Authoring Toolkit employs this technique of bundling both Insert Scenes and Theme Packs.

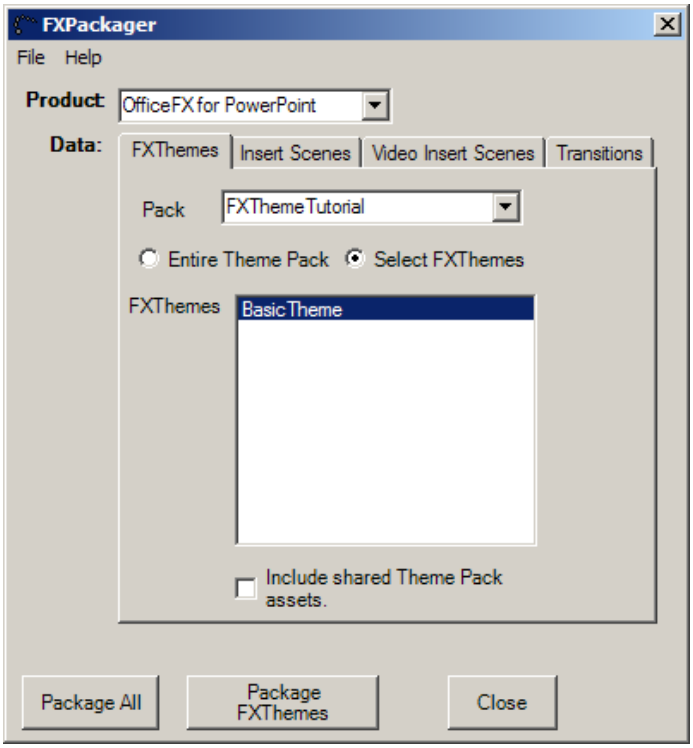


Fig. 42 FXPackager UI showing the Theme Pack called FXThemeTutorial and the FXTheme called BasicTheme selected for packaging.

With Theme Pack, FXTheme and optionally Insert Scenes selected, press the Package All button at the lower left of the FXPackager UI. This will bring up a standard windows file browser dialog that will allow you to name the resulting file and determine the folder into which it will be written. The resulting FXPackage file is in binary format with an .xml file extension. It can be sent to any Presenter user and installed using the Product / Install FXAsset utility on the Presenter toolbar. If Insert Scenes have been included, those will be placed into the \OfficeFX\Repository/Inserts folder.

FXPackager Caveats

There are several considerations in running FXPackager associated with which assets get pulled together and put into the resulting FXPackage file.

1) While the FXPackager does allow you to package all Themes in a Theme Pack, you can also select individual FXThemes from a Theme Pack for packaging. In both cases any asset files in the Theme Pack folder will be included, even if only individual FXThemes are selected.

2) Be cautious of name collisions. If you install a new FXTheme over a previously existing FXTheme of the same name, any files in the new Theme that have the same name as files in the older Theme will be over-written.

3) In general it is best practice to keep media files that will be used in a Theme Pack or FXTheme stored internally within those Packs and Themes. That will assure that the FXPackager locates and includes those assets in the FXPackage file that results.

4) Assets from the following folders are never packaged with a theme or insert scene, even if assets in those folders are being used:

- \Repository\Defaults
- \Repository\Plug-ins
- \Repository\Media
- \Repository\Media\Shaders

5) FXPackager will not package either the FXThemes that ship with Presenter or new FXThemes and Theme Packs that are purchased from Instant Effects. It is a tool that is provided to allow theme artists to create and deliver new FXThemes and Insert Scenes to customers or other users of Presenter within their company.

6) FXPackages for FXThemes always install to the same Theme Pack and FXTheme folder name as the one from which they were packaged.

7) Insert Scenes and Video Insert Scenes can either be packaged in the subdirectory they were created in, or the default Repository / Insert (or Video Insert) folders. A checkbox on the FXPackager interface provides that option.

8) FXPackages can be installed directly across a network or from devices like CDs or USB keys.

Advanced Topics

Section Overview

This section will introduce a knowledgeable FXTheme developer to techniques for generating more complex 3D assets for Presenter. These topics assume a detailed understanding of Presenter, the FXTheme building process and general concepts, as well as an in-depth knowledge of 3ds max. In some cases they will involve manual editing of FXML files generated by the FXML Export Utility program.

Topics covered herein are:

- Video Insert Scenes
- Using Video in Presenter Assets (including video with alpha)
- Deformations
- Multi-Material support
- FXTransition Packs
- Dealing with Z Sorting Artifacts
- Building Custom Aspect Ratio FXThemes
- Building FXThemes with User Color Control
- Triggering Insert Scenes with Action Sets
- Building Stereo 3D Content
- Miscellaneous other considerations

Example Files

Also included with this Instant Effects Authoring Toolkit are a set of example 3ds max files that illustrate how FXML scenes and assets that drive Presenter features are authored. Those files are in the FXThemeMaxFiles folder and now include:

- VideoInsertBillboard.max: Illustrates creation of Video Insert Scene.
- 4x1 VideoOnGeometry.max: A file that shows how to use.non-standard avi files as animated textures to create a wider variety of 3D Presenter assets.
- TwistDeformWithVideo.max: Illustrates the mechanism for authoring geometry that animates through the use of 3ds max deformers. This file produces an Insert Scene that features both deformation and animated texturing.
- Dragon_Character.max: Another example file that includes deformation. This one illustrates the use of character skinning deformers. It too generates an Insert Scene.
- TwistDeformTransition.max: Shows the method required for using deformations in scenes that will be employed in transitions.
- MultiMaterialsExample.max: Provides a logo example that employs more than one material per mesh, and can be exported as an Insert Scene

Video Insert Scenes

Video Insert Scenes are a significant feature of Presenter. They allow a user to select from a set of provided scenes and then map any video in the supported format onto those scenes. The supported video playback formats are:

- AVI
- WMV
- MPEG-1
- MPEG-2

Like all other aspects of Presenter, Video Insert Scenes are an open system. Custom Video Insert Scenes can be created in 3ds max. The FXML Exporter Utility will output those directly to the Repository \ Video Insert folder, where they become immediately available to Presenter users.

An example 3ds max file is provided that illustrates creation of a custom Video Insert Scene. It is a billboard set up in a simple terrain environment. The file is called VideoInsertBillboard.max and is with other example 3ds max files in the FXThemesMaxFiles folder. Bit maps used as textures in this file are in the Media sub-folder. As a Video Insert Scene in Presenter it looks like this.

Fig. 43 – The Billboard Video Insert Scene loaded into Presenter and sized to take up the full screen. The system's default video (with audio) is playing on the billboard itself.

Note: In this example the Video Insert Scene has been set to display behind PowerPoint content, allowing the slide title and the BasicTheme's foreground scene elements to appear in the frame as well.

Load the VideoInsertBillboard.max file into 3ds max and review the setup. It is standard in most every way. As is normal for any Insert Scene setup, the camera used is a Target Camera and the camera's target is at the center of the billboard. The camera target becomes the pivot point for interactive viewing of the Video Insert Scene in Presenter.

All of the aspects which set this file apart as being created as a Video Insert Scene are associated with the mesh called "Image Screen" that is set up to receive video. The distinctive elements of that mesh are:

- While it is designed to receive video as a Video Insert Scene in Presenter, no video is mapped to it in 3ds max. Instead it uses a simple white diffuse material with self illumination set to a maximum of 100. This self illumination setting assures that the video that gets applied to the mesh in Presenter will play with full color saturation and intensity.
- A special Property / User Defined entry is attached to the Image Screen mesh. That is: "VideoStandin=true"
- The aspect ratio of the Image Screen mesh is 4x3 and it has 1 to 1 (default) texture mapping. No custom UV mapping is applied. That means the Video Insert Scene that will be created is intended to play standard definition (4x3 aspect ratio) video. Since video is applied as a texture in Presenter's Video Insert Scenes, it is important to match the mesh size and texture mapping to the format of the video that it is being designed to play.

Now export the file as a Video Insert Scene. To accomplish this bring up the FXML Exporter Utility's scene export dialog and check both the "Output as Insert Scene" and the "This is a Video Insert" options. See Fig. 44. In this example we've also put the Video Insert in a subfolder called "Lessons".

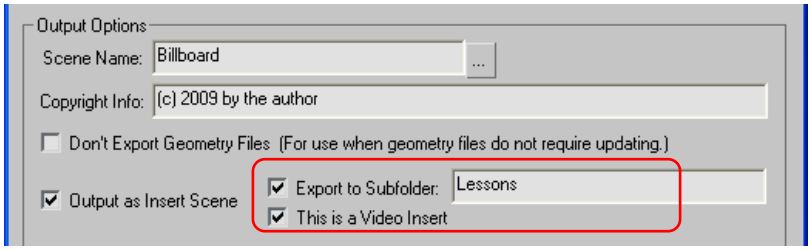


Fig. 44 – The Output Options section of the FXML Exporter Utility's scene output dialog with options checked for exporting a Video Insert Scene named Billboard.

After exporting, go into Presenter and add the new Billboard Video Insert scene to a slide. Remember it's in the Lessons subfolder if you've used that option. Experiment with size, position, and different viewing angles as well the Place Behind PowerPoint Content option. You can also now apply different videos. In particular, try applying both SD (4x3 aspect ratio) and HD (16:9 aspect ratios) format video. You will see that the SD video plays correctly but the HD video gets compressed horizontally. That is because all its 16x9 format pixels are being mapped to the 4x3 Image Screen mesh.

In the next section we'll discuss approaches to using video in more creative ways. Among other things, that discussion will include video with varying aspect ratios.

Using Video in Presenter Assets

The ability to use video as a design element in Presenter assets opens up many creative opportunities. Video can be used to create motion backgrounds. It can be used as animated texture in Insert Scenes. Even video with alpha (transparency) can be included. And with proper setup you are not restricted to using video with either SD or HD aspect ratios. You can include any custom aspect ratios you choose as a way to add interesting "lower third" or dynamic side bar type design elements.

In this section we'll deal with two aspects simultaneously. We'll create an insert scene that plays high aspect ratio (4X1) video that includes alpha (transparency) information. Before we get started a brief discussion of video compression / decompression (codecs) is in order.

Codecs and Video with Alpha

Most desktop video is compressed in order to reduce file size. With the huge growth in the desktop video industry, professional grade video codecs have become very efficient and high quality. They compress and decompress video rapidly and they do so without introducing significant visual artifacts.

Video with transparency however creates some special issues for decompression. In particular the key (alpha) channel usually carries the kind of sharp edges that causes artifacts in most compression algorithms. For that reason, most of the standard desktop video codecs (XVID, DiVIX, WMV, etc) do not support video with alpha.

One approach to compressing and decompressing video that does lend itself to treatment of video with alpha is "Run Length Encoding". That's a simple approach which only compresses sequences of pixels with exactly the same RGB values. Since video with alpha has large areas that have the same fully transparent values, this approach is effective in reducing file size. Since it changes nothing about the non transparent pixels, run length encoding is "lossless". It delivers a high quality result, and since there is really no complex math used to accomplish the compression, the decompression is simple.

Lagarith codec

The Lagarith Lossless Video Codec is a freely available (public domain) codec that delivers excellent run length encoding compression and decompression results. It can be located for download from a variety of sources on the Internet. Once installed, the Lagarith codec is recognized by most video applications (including Presenter). When you render to video simply choose the Lagarith codec from the codec options list that is presented in the video application you are using. Make sure to configure the codec to include the Alpha channel on output. Presenter's versions of those control option dialogs appear in **Figs** 45 and 46. While they will look different applications such as Adobe Premiere® or After Effects® or Sony Vegas®, their function will be the same.

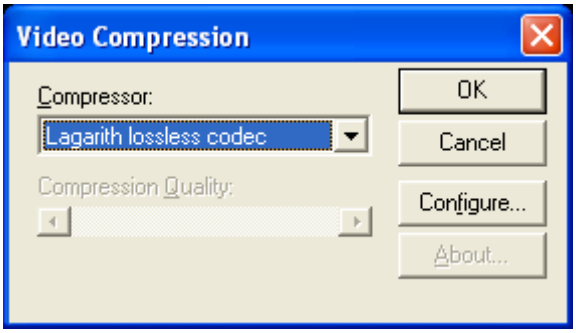


Fig 45 In Presenter's Publish / Video dialog check the "Choose" option in the Compressor section of the dialog. When you do you'll get this dialog. The pulldown will show all your installed codecs. Here the Lagarith lossless codec has been selected.

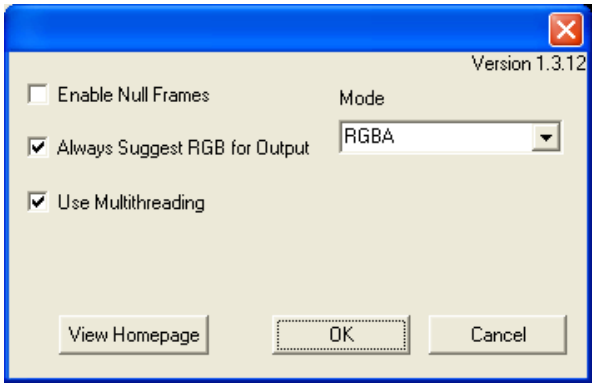


Fig 46 Press the "Configure" button in the Video Compression dialog and you'll get this dialog. To create video with Alpha, make sure that the RGBA Mode is selected. Also make sure to use a "Blank Background" variation in the FXTheme you choose.

If you have video that includes Alpha you can play it on one of the Video with Alpha Video Insert Scenes that are provided with Presenter. Those assets are set up to recognize the Alpha channel information. You'll see that Standard, Wide, and Letterbox versions are available.

You can test Presenter's Video with Alpha Video Insert scenes using the 4x1 alpha video from AE.avi file that you'll find in the FXThmemMaxFiles\Media folder that accompanies the Authoring Toolkit.

Note: The 4x1 alpha video from AE.avi uses the Lagarith codec. If that codec is not installed on your system, the video will not play. Completing this portion of the tutorial will require that you first download and install the Lagarith codec.

With the Lagarith codec installed, the Wide Video with Alpha Video Insert selected, and the "Interactive" FXPreview option set, you'll see this.

Fig 47 Video with alpha playing on Video with Alpha Video Insert in Presenter.

Note: the "Interactive" option must be selected under the FXPreview window in order to trigger video playback

Note that this video is properly displaying its alpha information. Where there is no text, you see through to Presenter's motion background.

Once you have the video with alpha playing properly in Presenter, play it with a standard video playback utility like the Windows Media Player. When you do that, you'll see a significant difference. The text will not be stretched vertically. Instead it will look more like this.

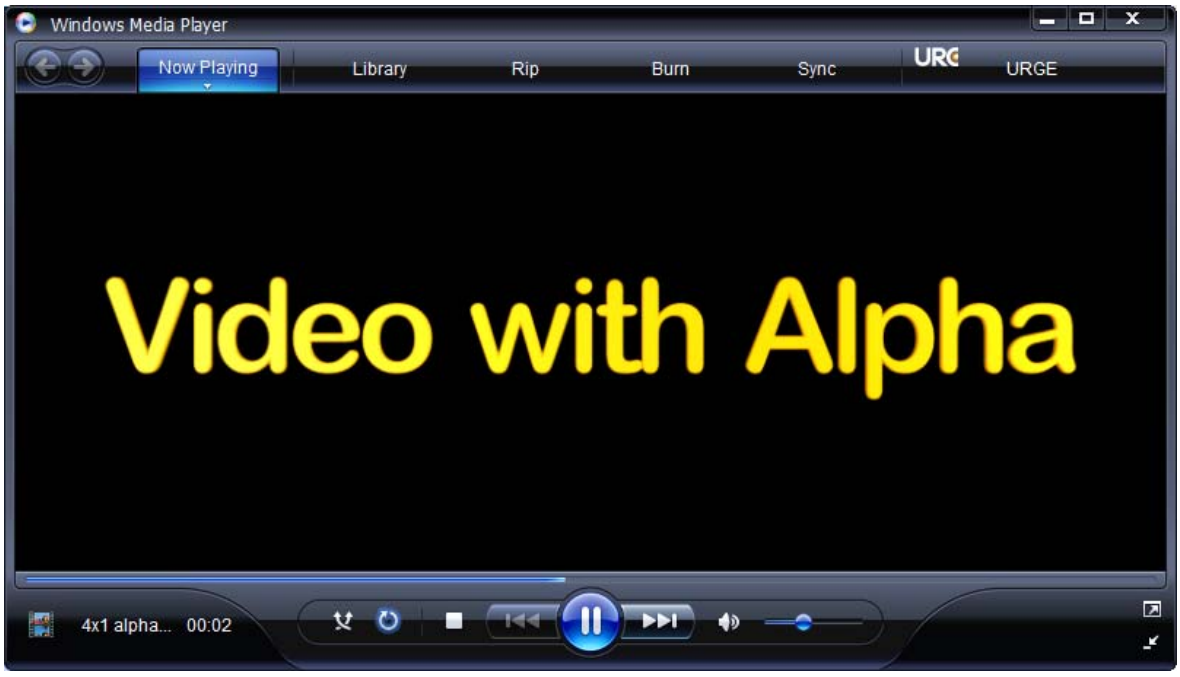


Fig 48 Shows the 4x1 alpha video from AE.avi file playing in the Windows Media Player.

The difference is due to the video's native aspect ratio. Per the file name, that ratio is 4x1. If you check the video in any video application you'll see that its resolution is 800 x 200 (a 4 to 1 aspect ratio). In the Windows Media Player, the video is playing at its native resolution. In Presenter it is being texture mapped to a Wide Video Insert scene that has been set up for the HD aspect ratio of 16 x 9. That is causing the video to be compressed horizontally. The rest of this particular exercise will focus on how to build custom scenes in 3ds max that properly play video of both non-standard aspect ratios and video that includes alpha information.

Note: Scenes developed using these techniques can be exported as custom Video Insert Scenes (using the techniques we discussed in the previous section). They can be used as Insert Scenes where the video is "hard wired" and not designed for changing by the user. They can also be incorporated into custom FXThemes as backgrounds or foreground scenes.

To get started, open the provided 3ds max file named 4x1 VideoOnGeometry.max. The mesh named video screen uses a material which applies the 4x1 alpha video from AE.avi file as a diffuse map. With that link properly established and with shading and texture display turned on, if you scroll forward in the time line you will see this.

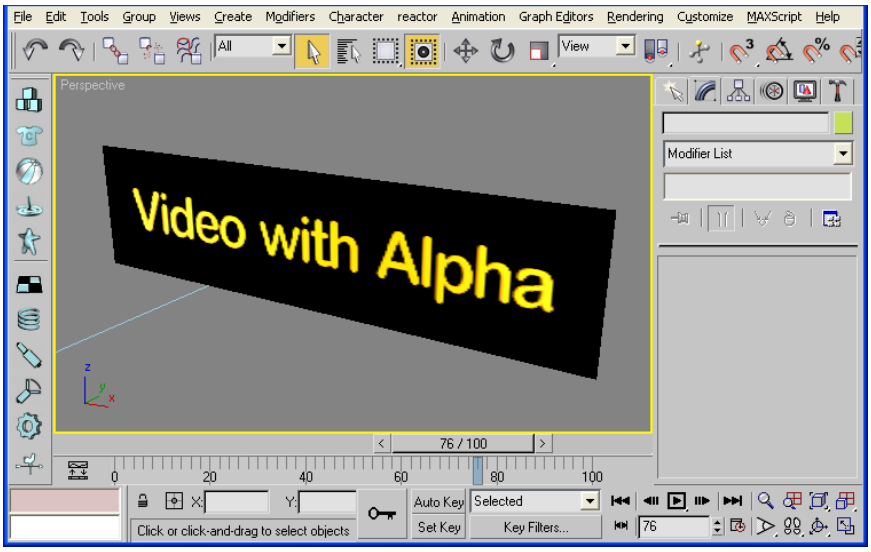


Fig. 49 3ds max showing the mesh setup for properly playing a 4x1 aspect ratio video.

There are several elements to note about this file.

- The rendering output resolution is set to 800 x 200. That determines the aspect ratio of the 3D viewport of the scene that will be created when it is exported by the FXML Exporter. If your intent is to use the scene as an Insert Scene or Video Insert Scene, matching the camera viewport to the dimensions of the video mesh will be important as it will give you most flexibility in sizing and placement within Presenter.
- The video screen mesh has the special Presenter Property / User Defined attribute of

  OpacityReferencesDiffuse=true

  This causes the same video (with alpha) that's been applied as a diffuse texture to be used to generate opacity information.

  Note: It is important that you use the above special User Property instead of simply applying the same video as an opacity map in 3ds max. That will work but it will cause the same video to be reloaded twice. That can cause the artifact of opacity map and diffuse map getting one or two frames out of step. The Opacity ReferenceDiffuse=true attribute causes the video source file to be loaded only once. The same stream gets used by both opacity and diffuse materials and the two always stay in step.
- The scene has no lights. That does not matter as the screen material applied to the video screen is set to self illumination value of 100. This assures that the video will play at full brightness without doing any 3D lighting calculation.

Now export as an Insert Scene (to the Lessons) folder. The FXML Exporter UI should like like **Fig** 50.

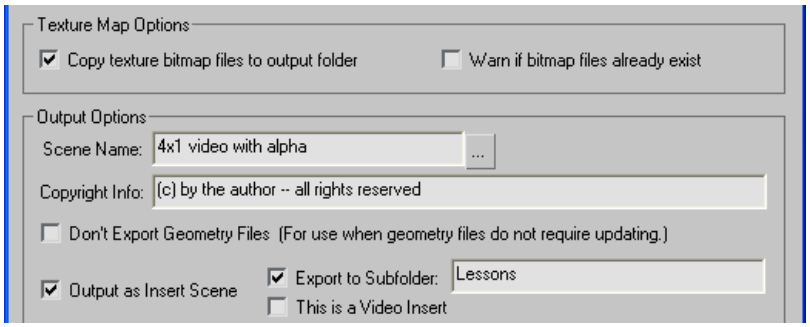


Fig. 50 FXML Exporter Utility set up to export the 4x1 VideoOnGeometry.max scene

If you then load the resulting Insert Scene in Presenter, you'll see that it plays at the proper size. It is no longer stretched vertically. It also respects the video's alpha channel. See **Fig** 51.

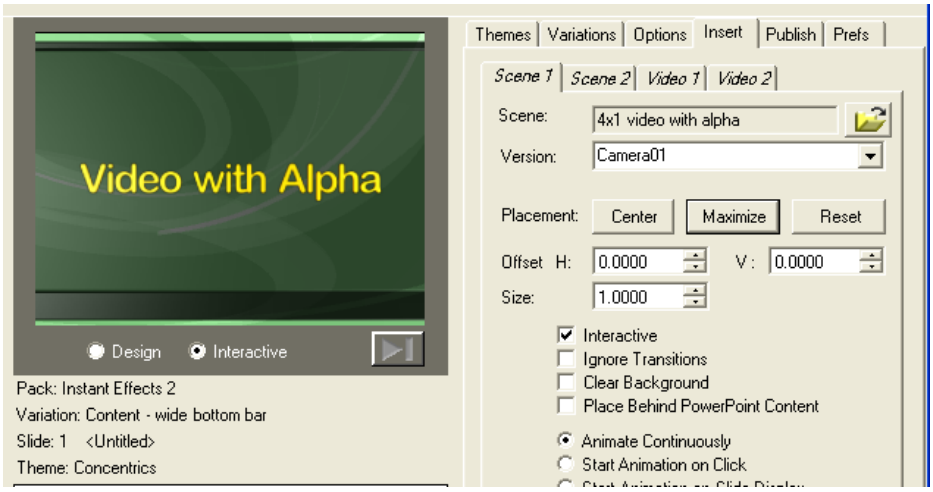


Fig. 51 Presenter set up to play the just exported Insert Scene that uses non-standard (800 x 200) resolution video that also includes alpha (transparency) information.

We've now developed the techniques required to use different aspect ratio videos as well as video that includes alpha (transparency) information. While the example we worked generated ah Insert Scenes it's easy to extrapolate how the same techniques would be used to create Video Insert Scenes of differing aspect ratios or scenes that employ video with alpha as part of FXTheme development.

Before concluding here are some additional notes on developing those custom 3D scenes that use video to control both diffuse color and opacity.

- 3ds max only recognizes AVI files and Quicktime (MOV) files as animated textures. The Presenter solution for streaming video is built on the Microsoft DirectShow API. It does not recognize Quicktime files, but it does recognize WMV, MPEG-1, and MPEG-2. Video in these formats however is not recognized by 3ds max. It's therefore easiest to work with AVI format video files when generating 3D assets for use in Presenter. Doing so allows you to see the result in both 3ds max and Presenter without doing file conversions or "hacking" fxml files.

  If you absolutely must use one of the video formats that 3ds max does not recognize you can use some stand in AVI file during the scene development in 3ds max. After you export, you can open the Scene.fxml file that is created in a text editor. Search for the XML block that references the stand in AVI file. It will look like this

  ```
  <Bitmap>
     <Video>.\stand in.avi</Video>
  </Bitmap>
  ```

  Change that reference to something like this.

  ```
  <Bitmap>
     <Video>.\yourDesiredVideo.wmv</Video>
  </Bitmap>
  ```

  This simply switches the video which will play back on the mesh you've designed. Make sure that yourDesiredVideo.wmv is located in the scene's Media folder so it gets found by Presenter.

- Video playback in Presenter does recognize audio.
- Video file size doesn't really matter. Because they stream, videos are never completely resident in memory so they don't consume large amounts of that resource.
- What does matter is bit rate. Videos with very high bit rates will begin to stutter and lose connection with their audio. Several other variables will affect this. The overall complexity of the FXTheme being used and the user's content will have an impact. Also important will be the performance characteristics of the computer's 3D graphics card. In general it is good to keep encoded videos to a bit rate below 8 Mbps. Lagarith encoded videos can run higher than that as their decoding algorithm is simple. Always test video asset playback on the system where they will be used.
- For viewing characteristics of video like bit rate, it is recommended that you have a good video diagnostics tool One that we find quite useful is called GSpot. It is available for free download from the Internet.
- Video asset management: The video files that are mapped to geometry must be managed just like any other media file within Presenter. For scenes other than Video Insert Scenes that incorporate a specific video, that video file must be placed into the appropriate folder to be recognized. The FXML Exporter Utility will include a video file that is used as a texture in the assets it exports to an FXTheme or Insert Scene folder

Deformations

Presenter provides a vertex cache system which allows for support of all the 3ds max deformation types. This includes Free Form Deformers, Parametric Deformers, and even character skinning deformations. The FXML Exporter utility exports the final position of each vertex in the deforming mesh on a frame by frame basis. As a result multiple deformations can be stacked in 3ds max. The end result of the stack calculation is what gets exported for each frame. By default, the position of every vertex in the deforming mesh is calculated every 30th of a second (the value can be controlled via User Defined Property). The resulting array becomes a vertex cache within the FXML scene file.

Deformations can be made to loop, to oscillate back and forth, or to just play once. That is controlled from within 3ds max with a User Defined Property as follows. In 3ds max, select the object that is deforming. In the Properties dialog, click on the User Defined tab and enter one of these choices (without the quotes):

- "Deformation=Loop" – causes the mesh animation to run repeatedly from first frame to last.
- "Deformation=PingPong" – causes the mesh animation to run repeatedly from first frame to last, then last frame back to first.
- "Deformation=Constant" – causes the mesh animation to play just once.

The 3dsd max file TwistDeformWithVideo.max provides a simple deformation example. It animates two twist deformers on a cube over 150 frames. In this example the Deformation=Loop option is employed. The cube has a video named skier.avi video to its 6 sides. That video is in the Media folder included with this Authoring Toolkit. Export this file as an Insert Scene, and then load it into a presentation in Presenter to view its playback characteristics.

Basic considerations for including deforming objects in your FXThemes and Insert Scenes are:

- Size matters: The deformation system is based on playback from a cache that contains the position of every vertex in every animation frame. Long animation on large meshes will produce very large vertex cache (vtx) files. In order for these deforming objects to play at high speed in Presenter, they must fit into physical memory. Be aware of complexity in the deforming mesh and overall length of your animation. When you use deformations make sure to test the results in actual show conditions with full content complexity.
- It is possible to reduce the vertex cache size by controlling the sampling frame rate at which individual verticies are generated. Do that with the User Defined Property of "DeformationFPS=number" (where number is the frame rate at which new vertices are generated. Note: there is no interpolation of position between vertices in Presenter. If you specify DeformationFPS=15 and set Presenter to play at 30 fps (default), then each vertex will play for 2 frames. This works well when deformations are small. Where deformations create larger vertex displacement, the result may be noticeably "jittery" animation.
- Today's CPUs are highly capable of dealing with large vertex cache files. Those cycle through memory highly efficiently and at extremely fast rates. On newer hardware vertex cache files of 40Mb set to play at 30 fps do so without any noticeable performance issues.
- A consideration for all deformations is that the deforming object has adequate internal mesh resolution to maintain a smooth surface as it bends, twists, tapers, etc. As an example, look at the mesh resolution in the simple cube in the example TwistDeformWithVideo.max file. It has over 4000 polygons to allow 6 flat sides to deform smoothly.

A second example 3ds max file is provided which illustrates the use of character animation and character skinning within Presenter. This file is Dragon_Character.max. It is a standard 3ds max character animation demonstration tool. Note its relatively short animation cycle.

Deformations in Transitions

Using deformations in transitions opens up some interesting creative opportunities, but there are a couple of special considerations. Primarily the deformation vertex cache does not explicitly carry information about how long the transition clip should be. In addition, Presenter needs an indicator that the vertex cache should be triggered for playback at the start of a transition clip. There are several specific measures that are required for using deformations in Presenter transitions. These are:

- Clip length must be conveyed to Presenter through inclusion of key frames on the deforming mesh. This is done in the standard way by saving keyframes for that mesh in one of the transforms of position, rotation, or scale. Note: Keyframes need not actually transform the deforming mesh. They only need to set the length of the clip. A transition clip targeted for 2 seconds should have keyframes on the deforming mesh at frame 1 and frame 60.
- The standard clip entry in the User Defined Properties of the deforming mesh is required. Choices are the usual Clip=P, Clip=R, or Clip=S, depending on which transform type has the keyframes that determine clip length.
- An additional User Defined Property must be added for using deformations in transition. This signals the deformation's vertex cache to start playing at the onset of the transition. The User Defined Property that must be added to those of the deforming mesh is:

  DeformationTransClip=true

The example 3ds max file named TwistDeformTransition.max is provided to illustrate the use of deforming geometry in transitions. After you have inspected it, export it as a scene file to the FXThemeTutorial Theme Pack's BasicTheme. To follow the example below, export using a scene named R2T Twist. That scene can be referenced as the Render to Texture scene in a new transition called R2T: Twist in BasicTheme. Set up for this transition in FXDesigner is shown in Fig 52.

Note: An additional creative option is exposed in this new R2T: Twist transition. That is the inclusion of a synchronized audio effect. In this case the audio file called Creaking_Door.wav is selected as the Transition Sound Clip. (see **Fig** 52). It plays as the transition is triggered. Creaking _Door.wav is provided as part of this Toolkit. It is in the Media folder below FXThemeMaxFiles.

Audio files are treated like any other media type in Presenter. In order to be accessed during presentation they must be in one of the four folders. If they are to be used by multiple FXThemes within the Theme Pack they should be in the Theme Pack folder or in a Theme Pack\Media folder. If they are to be used only be a single FXTheme, it is good practice to keep them with that theme. They can either be in the FXTheme folder, or in a Media folder within the FXTheme folder.

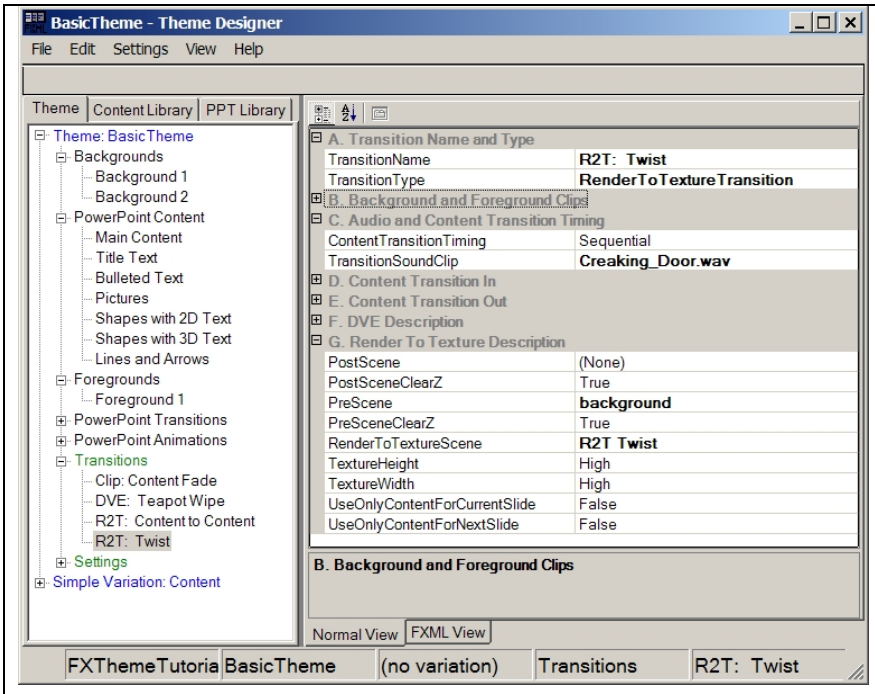


Fig 52 FXDesigner UI showing set up for new Render to Texture transition in BasicTheme. The newly exported deformation scene "R2T Twist" is referenced as the Render To Texture scene. The previously used "background" scene is referenced as the transition's PreScene. It is rendered first and composited behind the R2T Twist scene so that the background during the transition is not black. The transition can also be built without the background PreScene.

In the TwistDeformTransition.max file the following setup characteristics are worth noting. There are two separate meshes that are fit exactly to fill the screen of the scene's camera. These are the render to texture meshes that are designed to take the current slide's content away and bring the next slide's content into view. Both are animated using the same twist deformer setup. Two deformers are used for each mesh. One twists the top of its mesh 90 degrees at the onset of the transition. The second deformer twists the bottom of its mesh into place to conclude the transition. The twist deformer's centers have been moved to bottom and top (respectively) of each mesh to accomplish this offset.

Note also that both current and next slide meshes have position keyframes saved. Those keyframes do not actually change the position of the meshes. They are included to establish the duration of the transition. The Properties / User Defined entries for these meshes are as follows (without quotes):

- "R2TMaterial=NextSlide" – designation that tags this mesh to receive the content of the next slides as a texture. Note the lack of the "WithOpacity" suffix in this case. It's not necessary in this case as the UseOnlyContent option is set to False for both Current and Next Slide.
- "Clip=P" – the necessary clip declaration that directs Presenter to look to the key frames on this mesh to determine the time duration of the transition. In th TwistDeformTransition.max file, note the keyframes on the mesh's position transforms.
- "Deformation=Constant" – signals the vertex cache to play just once
- "DeformationTransClip=true" – tells the vertex cache to start playing at the beginning of the transition.

Multi-Material Support

Presenter supports meshes with more than one material. Similarly, the FXML Exporter Utilities have been extended to recognize the Standard Multi / Sub Object Materials of 3ds max. This streamlines the authoring of 3ds max scenes for export as FXTheme components and Insert Scenes. It also makes many more of the pre-built objects and scenes that can be purchased or obtained from other 3ds max users more directly useful in Presenter as those scenes frequently use Multi / Sub Object Materials. Now they can be exported as FXML files with significantly less custom preparation.

An example 3ds max file called Mult-Material Example.max is provided in the FXThemeMaxFiles folder included with this Authoring Toolkit. See Fig. 53. The bit maps used as textures and reflections in this file are included in the Media sub-folder.

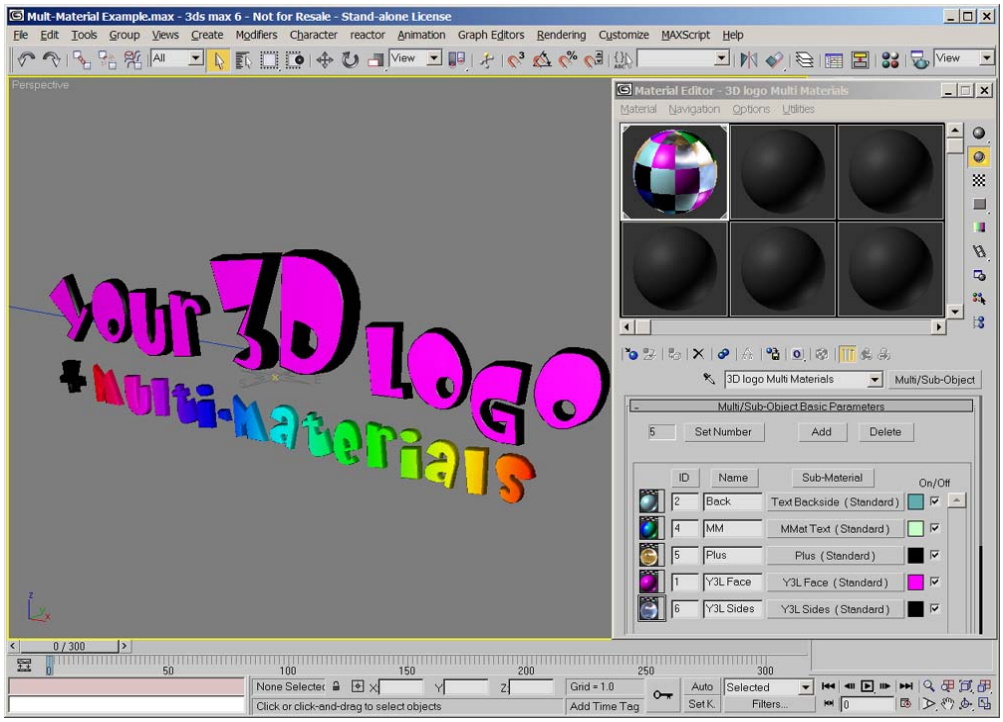


Fig 53 – The Mult-MaterialExample.max file loaded in 3ds max with Material Editor showing structure of the Multi / Sub Object material applied.

Load the Mult-Material Example file into 3ds max and inspect the material properties that are assigned to the 3D logo mesh in the scene. Export the file as an Insert Scene with the new FXML Exporter. As it creates the new Insert Scene folder and places all the necessary scene assets therein, it will also gather and export all the bit map files required for the Multi / Sub Object material that is applied to the logo in 3ds max.

Note: A limitation on the use of Multi-Materials is that they do restrict the use of visibility animation.. It is not possible to animate the visibility of objects that use multi materials where opacity is not 100% - even if the mesh whose visibility is being animated does not use the non-opaque material. The two values become confused. Where visibility animation is required the use of multi-materials is not recommended.

FXTransition Packs

Presenter supports three types of Transitions. It recognizes most types of PowerPoint transitions. It also recognizes the custom transitions that are part of a specific FXTheme. It also recognizes Transition Packs. Those are packs of transitions designed to work with any FXTheme of a particular type. The Standard Aspect and Wide Aspect transition packs that come with Presenter are examples. The transitions in those pack are designed to work with any Standard Aspect (4:3) or Wide Aspect (16:9) presentation. At those aspect ratios they can be added to any presentation and work with any FXTheme. These new Standard transitions are packaged into FXTransition Packs.

The open architecture of Presenter allows new FXTransition Packs to be created by FXTheme artists. The process involves building a dummy FXTheme from which the FXML and transition assets are extracted. We'll use the 3ds max files called TwistDeformTransition.max and background.max that are provided with this Authoring Toolkit to illustrate the process. Steps are as follows.

Open FXDesigner and create a new FXTheme in the FXThemeTutorial theme pack. For the purposes of this tutorial we'll call the new FXTheme "BasicTransition". Number Background Scenes should be set to 1. See Fig. 54.

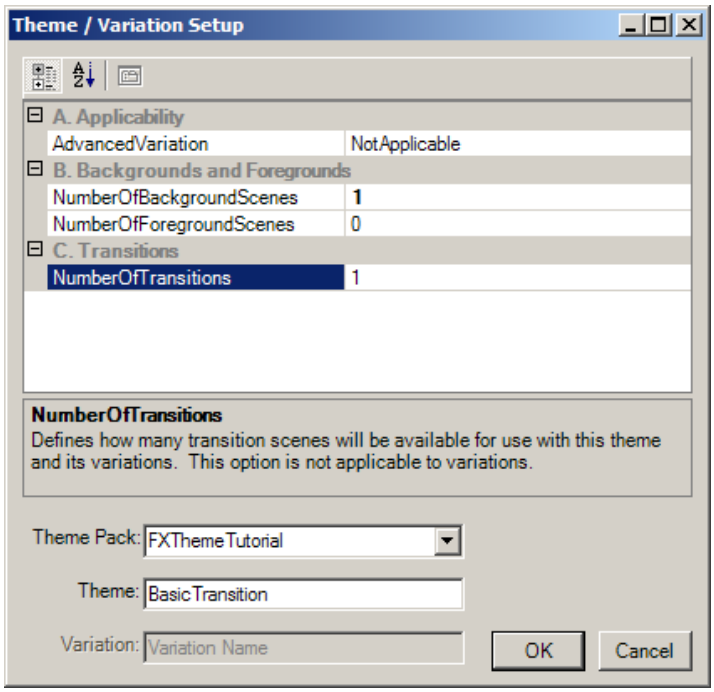


Fig. 54 – FXDesigner user interface showing settings for creation of a new Theme Pack named TransitionPack and FXTheme named BasicTransition.

Number of Background Scenes is 1. Number of Transitions is 1.

Note: Numbers of Transitions, Background, and Foreground Scenes can be changed at any time by clicking on the FXTheme name in the top of the left column in FXDesigner.

Clicking on File / Save creates the new Theme Pack and FXTheme folders in Repository\PowerPoint.

Now load the file TwistDeformTransition.max into 3ds max and re-export it to the new BasicTransition FXTheme folder you've just created with FXDesigner. Use the name "R2T Twist" in exporting the scene. Make sure you export to the FXTheme folder, not to the Theme Pack folder.

Next, load the file background.max into 3ds max and re-export that as well to the new BasicTransition FXTheme folder. Use the name "background" in exporting the scene.

In FXDesigner verify that the new TransitionPack and BasicTransition are still selected and click on View / Refresh Content (Ctrl R). That will register those new scenes and assets for use with FXDesigner. Next link the newly exported scenes to their respective elements of the FXTheme just as you did in the previous tutorial of building the BasicTheme.

Click on Background1 in the left column of FXDesigner. In the right column select “background” as the FXML Scene. See Fig 24 (page 44).

To build the transition, you may first have to go back and re-enter the number of transitions for the FXTheme. Do that by clicking on the FXTheme name at the top of the left column. Enter the number 1 for NumberOfTransitions and hit “OK”. Now select Transition1 in the left column and follow the same steps as before in creating the R2T Twist transition. See Fig. 43 (page 75).

For this transition we’ll go for just the simple twist. See Fig. 55.

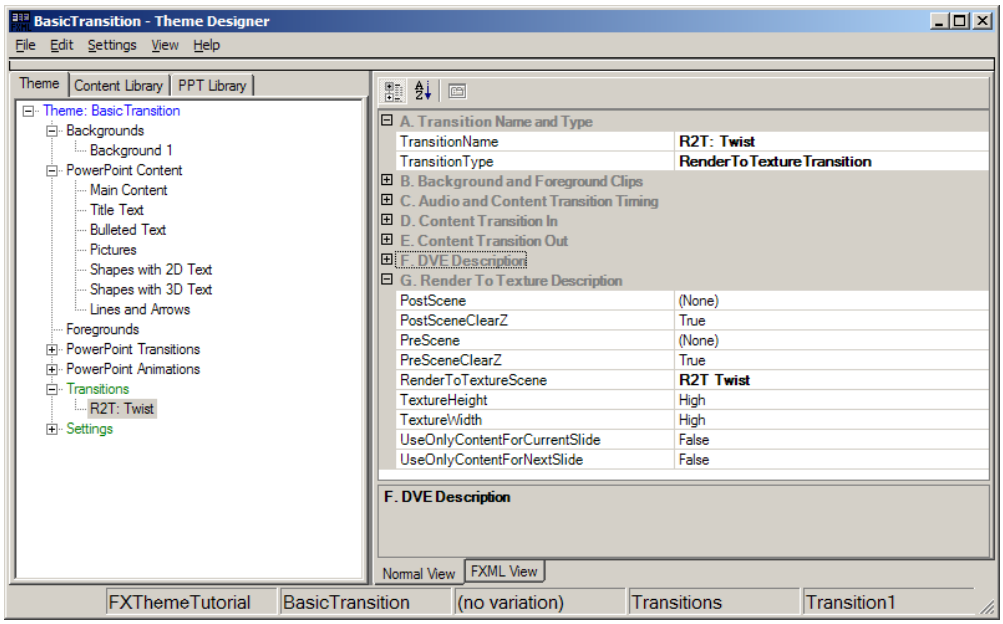


Fig. 55 – FXDesigner settings for simple R2T Twist transition. Note that we have not linked the creakingDoor.wav file and have not linked in the background scene as a PreScene.

The reason for not linking in the background scene as a PreScene is that this FXTransition is intended for generic use with any FXTheme. We do not want to see the background scene from the FXTheme Authoring Tutorial with its slowly rotating rings in the background when this transition is used with other FXThemes. When this transition is used we’ll see black in the background.

The final step involves making a copy of the BasicTransition FXTheme folder you’ve just created and placing it into the PowerPoint \ Defaults folder using a specific naming convention. It requires navigating to the OfficeFX \ Repository. Here are the steps

- Go to OfficeFX\Repository\PowerPoint\TransitionPack. This is the Theme Pack folder you’ve just created. Make a copy of the BasicTransition folder that is there.
- Go to OfficeFX\Repository\PowerPoint\Defaults. Move the copy of the BasicTransitions folder here. If there was a Media sub folder that’s used, make sure that is copied too.
- Rename the folder to Transitions.Basic. You’ll see two other folders there with similar names. Those are Transitions.Standard Aspect and Transitions.Wide Aspect. These are the standard transitions folders that ship with Presenter. You’ll now have a third standard transitions folder called “Basic”. The transitions in all these folders are designed to be used with any Instant Effects presentation and any FXTheme of appropriate aspect ratio.

Now verify that your new Basic FXTransition Pack is available as a standard OfficeFX transition, start up Presenter and click on the standard transitions pulldown. See Fig. 56.

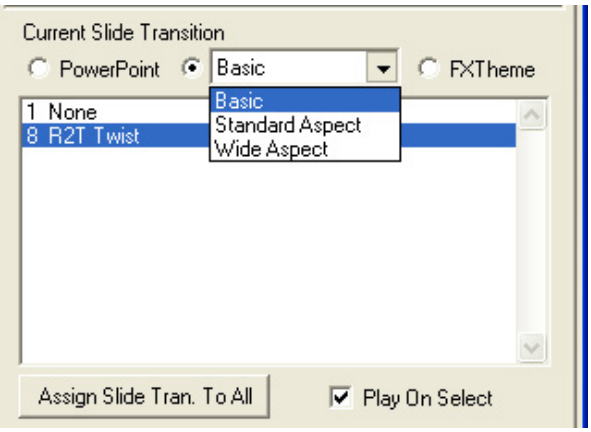


Fig. 56 – OfficeFX Variations Tab user interface with the Transitions portion of the dialog showing the new Basic Transition Pack and R2T Twist transition as available for selection.

Note: In the case of the Basic Transition Pack we've just created, the appropriate aspect ratio is 4:3. To see the visual artifacts created when mismatched aspect ratios are used, change the presentation's Page Size in PowerPoint, then use the new R2T Twist transition. You'll see the screen snap to a 4:3 aspect ratio during transition, then snap back to whatever aspect ratio you've set by changing the presentation Page Size

Hint: You may wish to remove the BasicTransition theme from the FXThemeTutorial once you've done this exercise. If you leave it there, it will show up as a selectable FXTheme in OfficeFX.

Dealing with Z Sorting Artifacts

Presenter is a real-time rendering system. To achieve a high rendering frame rate it has to take some numerical short cuts to the way it processes 3D scenes. One of those is a highly optimized way of determining what meshes and polygons are in front of other meshes and polygons. This process is called "Z Sorting".

The shortcut that Presenter takes is to use the mesh centroid (average position of all the mesh X,Y,Z values) as the Z depth for all polygons in a mesh. The Z depth is defined as distance from a polygon to the camera. The Z depth of a polygon determines whether it is in front or behind other polygons.

The centroid assumption shortcut allows real time Z sorting algorithms to run through large polygon lists very quickly. It works very well for opaque meshes but when transparency is introduced things get more complicated. A mesh may have both transparent and opaque polygons. The rendering algorithm needs to differentiate those and ignore the transparent ones. With the "centroid" assumption their absolute position may not be adequately accurate. This can cause Z sorting "artifacts". Those show up as either the wrong set of polygons rendering in the foreground, or flashing of as one set of polygons gets swapped sequentially in front or behind others. This problem is common in all real time rendering systems such as computer games. It's also called "Z fighting".

To combat Z sorting artifacts, Presenter allows the 3D artist to "steer" Z sorting with a special User Specified Property that forces an offset to the calculated centroid of any mesh. Here's how it is used.

Lets say you want to generate a globe model where the oceans are transparent. The continents appear in one color as they rotate by the camera. As the rotate to the back of the globe you see their under sides in a different color. That might look something like this.

Fig 57 Two identical coincident spheres rendered in 3ds max. Both use a material that employs the same cylindrical projection earth image as an opacity map. The outer facing globe's material has a green diffuse color. The inner facing sphere has reversed normals. It's material has a dark orange diffuse color.

3ds max uses a software rendering algorithm that Z sorts individual polygons (instead of using their mesh centroid). It is not fast enough to run in real time but it also has no problem differentiating which polygons are supposed to be in front.

To illustrate the issue further and provide an example of the solution a 3ds max file named ZOffset Globes.max is included in the FXThemeMaxFiles folder of the Authoring Toolkit. If you open that file now you'll see the above setup. Note that the two spheres are identical in size. The one named "inside" has its normals reversed. If you export that as an Insert Scene and load it into Presenter you'll get a very similar – as in **Fig** 58.

Fig 57 An exported Insert Scene where the inner globe mesh includes a User Specified Property that offsets that mesh's centroid away from the camera.

If you look at the User Defined Property that has been added to the sphere mesh named "Inside" you'll see that it carries the following attribute.

"ZOffset=10"

That causes Presenter to offset the centroid of that mesh away from the camera by 10 units. Those units are in whatever units you have set in 3ds max. Note that this offset is used for purposes of Z sorting ONLY. It does not cause any actual transformation of the mesh to another position.

The way Z sorting artifacts show up vary from computer to computer. They are a function of how the computer's graphics subsystem resolves Z depth. You can experiment on your own system by exporting the provided globe model without the ZOffset=10 User Defined Property. On the author's computer that results in an Insert Scene where the inside mesh gets rendered on top of the outside mesh. The result appears in **Fig** 58.

Fig 58 The same model with the ZOffset=10 User Defined Property removed. Not that the orange under sides of the continents that are on the back side of the globe are rendered on top of the front facing continents.

Replace the ZOffset=10 User Defined Property and re-export. You'll again get a correctly displayed version of the model where the front facing continents are always on top of those that have rotated around to the back side of the globe.

Building Custom Aspect Ratio FXThemes

In many of the exercises so far there has been close attention paid to aspect ratio. This is because most all forms of digital media are sensitive are to issues of aspect ratio. This is certainly true for video. The broadcast industry is part way through its evolution from Standard Definition (SD) video that carries a 4:3 aspect ratio to High Definition (HD) video with its wider 16:9 aspect ratio. Anyone who watches television is familiar with the compatibility issues associated with watching SD content on and HD (flat screen) display and vice versa. Images compress or expand off the screen.

It's also true of both PowerPoint and Presenter. If you change the Page Size of a PowerPoint file (in Page Setup) from the default 4:3 aspect ratio to something wider after it's been populated with content all the shapes and pictures will stretch horizontally.

That problem is exacerbated with Presenter as it's a full time 3D display system. You can think of an FXTheme as the staging of a live theater event. You will see numerous artifacts if you arbitrarily assign an FXTheme that's not designed for wide aspect ratio to a PowerPoint file that has a widescreen page size. Render to Texture transitions will snap to the wrong page size as they animate and you may also see details on the margins that were not meant to be displayed. Following the live theater metaphor, it's like you've pulled the curtains back further than you were supposed to.

For a quick demonstration of this, change the Page Setup / Size of the PowerPoint file you've been using to run the tutorials in this Toolkit. You might try something odd like 16" wide by 10" high. That will perfectly fill the screen on a notebook with screen sizes of 1440 x 900, 1680 x 1050, or 1920 x 1200, but it's neither a standard (4:3) or widescreen (16:9) aspect ratio. Instead it's 16:10.

Once you've made that switch, try out the Basic Theme you've been developing. The first thing to test is the transitions. You'll note several different things.

- First – the Cross dissolve: Same Variation – 1 sec transition works fine as long as you are not changing variations. That's because it only affects your PowerPoint content. It does not dissolve any of the FXTheme assets. That's what restricts its use to transitioning between slides that employ the same Variation.
- Second – the DVE: Teapot Wipe transition pops onto the screen at the left and doesn't make it all the way off the screen to the right before it ends. Following the live theater metaphor, this is a case of the curtain being pulled back too far. The remedy is simple. The animation for that transition would need to be modified so that those elements started and ended their motion paths off of the the new wider screen. To verify that, you'd simply set your camera resolution in 3ds max to match the desired aspect ratio of your presentation. Then edit the animation of the transition elements so they exit and enter cleanly.
- Third – both the R2T: Content to Content and R2T: Twist transitions pop to their designed 4:3 aspect ratio as they begin. When they are done, the screen pops back to the 16:10 aspect ratio that you've set temporarily. Since R2T transitions use texture mapping to snap a picture of your slide, then map that picture onto some 3D geometry – they are HIGHLY sensitive to aspect ratio. To use R2T transitions, the aspect ratio of the presentation must match the aspect ratio for which the R2T transition was designed EXACTLY. For that reason, it is not recommended that you use R2T transitions when building an FXTheme that is designed for use at a variety of aspect ratios.
- Finally, it's instructive to look at how the Basic Theme performs with a really wide aspect ratio presentation. Note that you'll need to be running either Event Presenter or Master Presenter to try this next test. It would be the type of setup you might use with Event Presenter and its ability to run a single ultra wide display out to multiple overlapped projectors. For example you may want to try an aspect ratio of 3:1. To do that click on Page Setup and set your Page Size to 12" wide by 4" high. When you go into Presenter you'll see a variety of issues, some obvious and some more subtle **Fig** 59 provides an example.

Fig 59 The Basic Theme running with a PowerPoint file set to 3:1 aspect ratio. At this extreme wide ratio the background scene does not look right. The camera reveals too much of it on left and right.

Also the Main Content scene's camera is too close to the PowerPoint text. As a result you see too much of its sides and bevels. It becomes difficult to read.

Once again however – Presenter is an open system. All these issues can be addressed by building FXThemes that are set up either to handle a variety of aspect ratios, or are designed specifically for some non-standard aspect ratio.

Instant Effects Presenter is restricted to 16:9 aspect ratios and below, but Event Presenter and Master Presenter allow users to deliver presentations without any restriction on aspect ratio. The new "Instant Effects Ultra Wide" Theme Pack is set up for presentations of up to 3.5 to 1 aspect ratio. If you are running either Event or Master Presenter and assign the "Deep Horizon" FXTheme from that pack you'll get something more reasonable. It will look like **Fig** 60.

Fig 60 Same PowerPoint content and Page Setup as Fig 50, but with Deep Horizon ultra wide FXTheme assigned. Note that theme color has been modified from default blue to make the background more obvious.

This section will cover some of the design issues associated with building FXThemes of these wider aspect ratios. In general those really just involve making sure to design both background and transition scenes so that they are capable of displaying a PowerPoint file with Page Setup / Size of the aspect ratio you wish to use. There are however a few special nuances and considerations.

Using "Safe Frame" to test ultra wide content in 3ds max

This is a basic technique that applies in creating virtually all custom 3D assets for Presenter. It's just a bit more explicit when aspect ratios begin to vary from the 640 x 480 (4:3) aspect ratio that is default for 3ds max. To illustrate, we'll use the example we referenced earlier of making the DVE transition of your Basic Theme compatible with the non standard 16 : 10 aspect ratio that's common on many notebook computers.

To do this start up 3ds max and load the DVE transition.max file that you've used previously. If you bring up the Render Scene menu you'll see that the rendering resolution is set at 640x480 – a 4:3 aspect ratio. Change that resolution so that matches that of a standard 16x10 display – e.g. 1680 by 1050.

Note: It's not the output rendering resolution that's important here as we are not going to actually render anything. It's the rendering aspect ratio that's key. That's how the viewport for your 3D scenes is determined in Presenter. It also determines what content will be visible and what will be "offstage".

Go to a single view of your scene and look display the scene through it's Camera (vs. the Top, Front, Right orthographic or Perspective views). Now scrub the timeline back and forth. If you stop at some frame where the teapot and the stencil plate are halfway across, the max interface should look something like Fig 61.

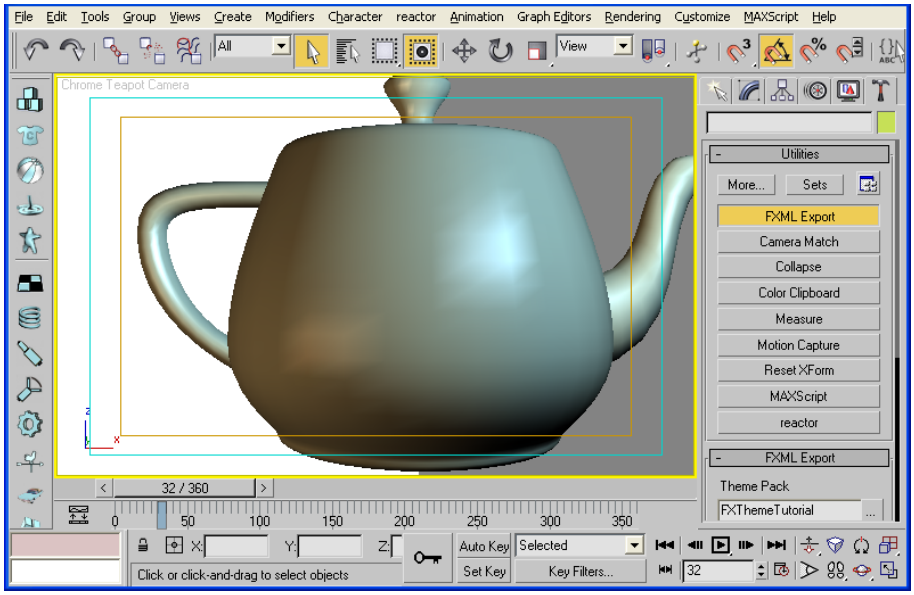


Fig 61 3ds max interface with DVE transition.max file loaded. Just the single camera view is displayed.

Note that the Frame Safe guides have been turned on. Those display the action safe and title safe areas. They also show the exact dimensions of the camera's rendering area.

As you scrub the animation back and forth you see that with the new camera aspect ratio of 16:10, both the teapot and the stencil plate do start out of frame (off camera). At the end of the animation, the teapot is fully out of view and the stencil plate is fully in view. This means that no adjustment to the animation really needs to be done. It just needs to be re-exported with the new viewport dimensions.

If you redo those 2 exports now and re create the DVE Teapot Stencil and DVE Teapot Visual Representation scenes, then play the transition in Presenter you'll see that it now behaves correctly. The Teapot slides in from off camera instead of suddenly appearing at the left margin of the slide. To end the transition, the Teapot slides fully off the frame and the stencil that reveals the new slide's content covers the frame completely.

Main Content Scene

The next "fix" we might consider really does not apply for a theme that's so close to a standard aspect ratio. It only became apparent when we looked at the file with an ultra wide aspect ratio. When we did that we noticed that the text at the far left and right margins was showing too much of its sides. This is because the standard lens setup in 3ds max is a 50° field of view (FOV). With that FOV you get significant perspective distortion at the margins of an ultra wide frame.

The remedy is to pull the camera back in the Main Content scene and go with more of a telephoto lens. That will narrow the FOV and reduce perspective distortion at the left and right margins of a wide aspect frame.

Doing this right will be an iterative process. The Main Content Scene is what determines final positioning of your PowerPoint content in frame. If you want to provide an exact match between positioning of that content in PowerPoint and where it's located in Presenter, you'll have to iterate that camera position and FOV settings until that is achieved.

Hint: To do this quickly use a test slide in PowerPoint that has square AutoShapes located right at the four corners. Export a new Main Content scene with the camera pulled back and a narrower FOV set. Bring up that slide in Presenter and see where those AutoShapes appear. If they are off screen, pull the camera back further in 3ds max and re-export. If they are too far towards the center of the view in Presenter, then move the camera forward in 3ds max and re-export. Repeat this process until those 4 AutoShapes are right in the 4 corners of the viewport in Presenter.

Wide aspect transitions

For the most part with wide aspect themes you'll likely do DVE transitions. Those will be more practical with so much horizontal screen space. From the above discussion you'll know to use the technique of matching your 3ds max camera's rendering resolution to the desired aspect ratio of your presentation. For example if you want to build an FXTheme for a 3.5:1 aspect ratio presentation, then set your rendering resolution in max to something like 1050 x 300.

Also remember to use the Safe Frame guides as you view the results of your animation in 3ds max. Those will show you whether the animation elements in your transitions are completely on or off the frame of your ultra wide camera's view.

Lastly, for ultra wide transitions you'll want to consider timing and direction. If you build wipe transitions that move horizontally, they'll have to cover a lot of screen if your aspect ratio is very high. You may want to slow them down a bit so they are able to traverse that wide area in a visually pleasing manner. By the same token, if you use similar wipe elements but move them vertically there will be far less screen space to cover. In order to make the two transitions compatible from design perspective they should move at roughly the same speed. In the vertical case that means the transition will be shorter.

Perfect R2T cross dissolves

R2T cross dissolves have advantages in Presenter. They do allow you to transition between slides that use different Variations. They also allow you to transition between slides that use Insert Scenes. In both those cases the Same Variation Cross dissolves fail. They only dissolve PowerPoint content so the different Variation elements and Insert Scenes pop into or out of frame.

The problem with R2T cross dissolves is that they are completely aspect ratio dependent. They only work properly if the aspect ratio of the presentation matches EXACTLY the aspect ratio for which the R2T cross dissolve was created.

In addition, to get a perfect match an exact mathematical formula must be used to set up the R2T scene in 3ds max. This section will show you how to achieve that precise setup.

To cover this technique we'll build an R2T cross dissolve for a presentation with an aspect ratio of 3.5 to 1. An example 3ds max file named R2T 3.5 x 1 cross dissolve.max is provided but more important is the general mathematical approach. Understanding that will allow you to generate well matched R2T cross dissolves for themes designed for any aspect ratio.

The basic math elements are these. You need to create "current slide" and "next slide" R2T meshes that exactly fill the viewport of the aspect ratio for which you are building. To do that you use the techniques of trigonometry. In particular for our example, we are seeking to create a cross dissolve for a 3.5:1 aspect ratio. Here are the steps.

First build current slide and next slide meshes that are 350 units wide by 100 units high. Add the necessary visibility clips and R2TMaterial User Defined Properties. Both meshes should be located at the origin and oriented normal to the max Y axis. That has been done in the example R2T 3.5 x 1 cross dissolve.max file. Next create a target camera with target at the origin and pull it way back along the negative Y axis.

Hint: Use max grid snapping as you create and locate this camera. That will facilitate placing both it and its target right along the Y axis.

For now just pull the camera back until the R2T meshes fill its view. Next change max's rendering resolution to some 3.5 to 1 aspect ratio. The example file uses 1050 by 300.

Now the challenge it to configure the camera so those meshes fill it exactly. First narrow down the FOV to something like 4° That will create a very long lens – something over 500 mm. Again pull the camera back so that the R2T meshes approximate fill the screen. Now for some trig. We'll use the known size of those R2T meshes and the known FOV to solve for what the precise distance should be between camera and the origin where meshes are located. The setup will look something like **Fig** 62.

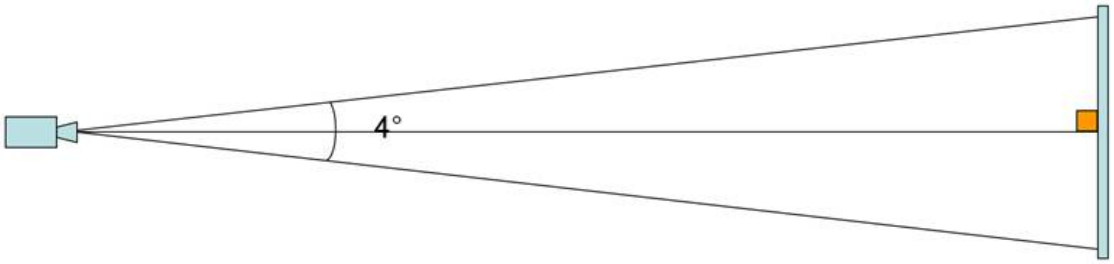


Fig 62 Trigonometric setup for calculating 4° FOV camera distance from 350 unit wide meshes

The "half angle" in this setup is 2°. The distance we're looking for in "trig terminology" is "adjacent" to that 2° angle. The distance we know is "opposite" the 2° angle. That distance is 175 units (half the width of the 350 unit R2T meshes. In trig, the tangent of the 2° angle = opposite / adjacent. Solving for the adjacent we get 5011.344 units. Set that as the distance of the camera from the origin along the negative Y axis and export the R2T 3.5 to 1 cross dissolve scene. If you then set up a new R2T transition in FXDesigner and reference the just exported FXML scene, you'll have a R2T cross dissolve transition that works well in a 3.5 to 1 aspect presentation.

Note: With the variable of differing screen resolutions you may still see a slight sub pixel shift during the resulting R2T cross dissolve transition. That amount of shift is unavoidable with the R2T technique.

Building FXThemes with User Color Control

You've probably noticed that since the launch of Presenter, all FXThemes have been provided as assets that can be color adjusted by the end user. This makes generic assets much more broadly useful. They can be tuned to the colors used in customer marks, logos, and product badges. In this section we'll cover ways in which you too can make assets that are color adjustable by your Presenter users.

The technique for adding color control is straightforward. It relies on the usual method of adding special User Defined Properties to the meshes that you want the user to be able to control. The syntax of those User Defined Properties is:

DiffuseColorFromSwatch=number

The number of colors a user can control is limited to 4. The "swatches" referred to above in the User Defined Property are the ones that appear with color controllable themes in Presenter's Prefs interface.

As an example, a max file named background.with color control.max is available in the FXThemeMaxFiles folder that accompanies this Authoring Toolkit. That file is a different version of the background scene that's used in the Basic Theme. Open the background.with color control.max file now.

The changes to note are these:

- The curved plate background plate element has a **DiffuseColorFromSwatch=1** User Property
- The 3 torus rings have **DiffuseColorFromSwatch=2** User Properties
- The torus rings are also less transparent than in the original background scene. That's just to allow their user defined color in Presenter to be more apparent.
- The texture maps on both those mesh elements have been removed. Those were noise maps but they also controlled the diffuse color. We are now going to allow the user to control those colors

Once you've inspected the file you can export it to the Basic Theme folder using some new file name.

Note: It's not necessary to change the names of the meshes in this case. The same meshes will work for both the original Background scene as well as the new color control background.

One more step is required to introduce color adjustability in the Basic Theme. That's accomplished in FXDesigner. Start that application now and load your Basic Theme. If it was already open when you did the above export, click on View / Refresh to make sure the application is aware of the new asset files in the Basic Theme folder.

For purposes of this exercise we'll just swap the original background.fxml file with the new color control background.fxml file you just created. To do that select Background 1 in the left column. In the FXML Scene category on the right, click the pulldown and select color control background. The FXDesigner should look like **Fig** 63.

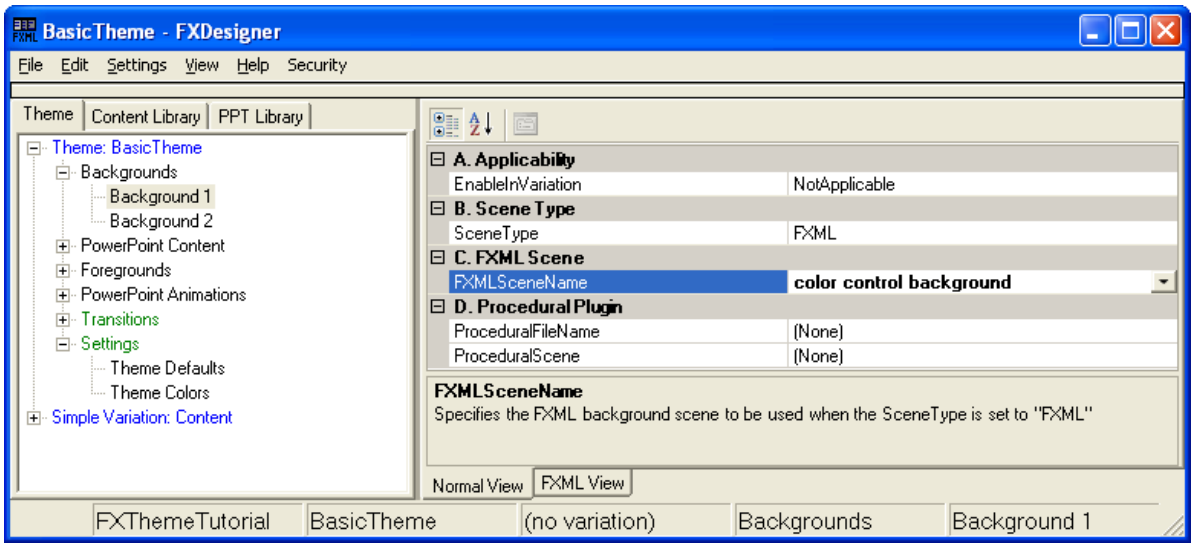


Fig 63 FXDesigner with new color control background FXML file loaded as the Basic Theme's Background 1 Scene.

Now there's one more capability that has to be enabled. Click to expand the "Settings" category in the left column of FXDesigner, then select the "Theme Colors" option. On the right side of the interface enter the number 2 in the NumThemeColors attribute (see **Fig** 64). This number corresponds to the two swatch colors you've added as User Defined Properties in 3ds max.

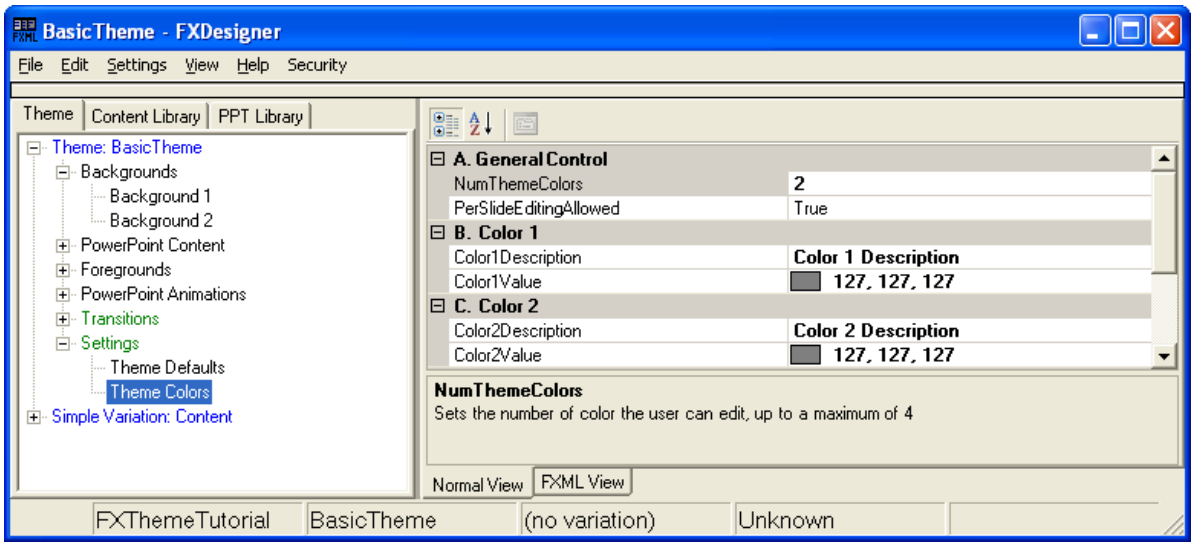


Fig 64 FXDesigner with Basic Theme set to take user control over 2 colors.

Now save the Basic Theme and restart PowerPoint. When you load the Basic Theme it will start with the grey tones as it did before, but if you click on the Prefs tab in FXEdit you'll see that you know have color control swatches. If you adjust those you'll see the changes take effect in the FXPrevew window as in **Fig** 65.

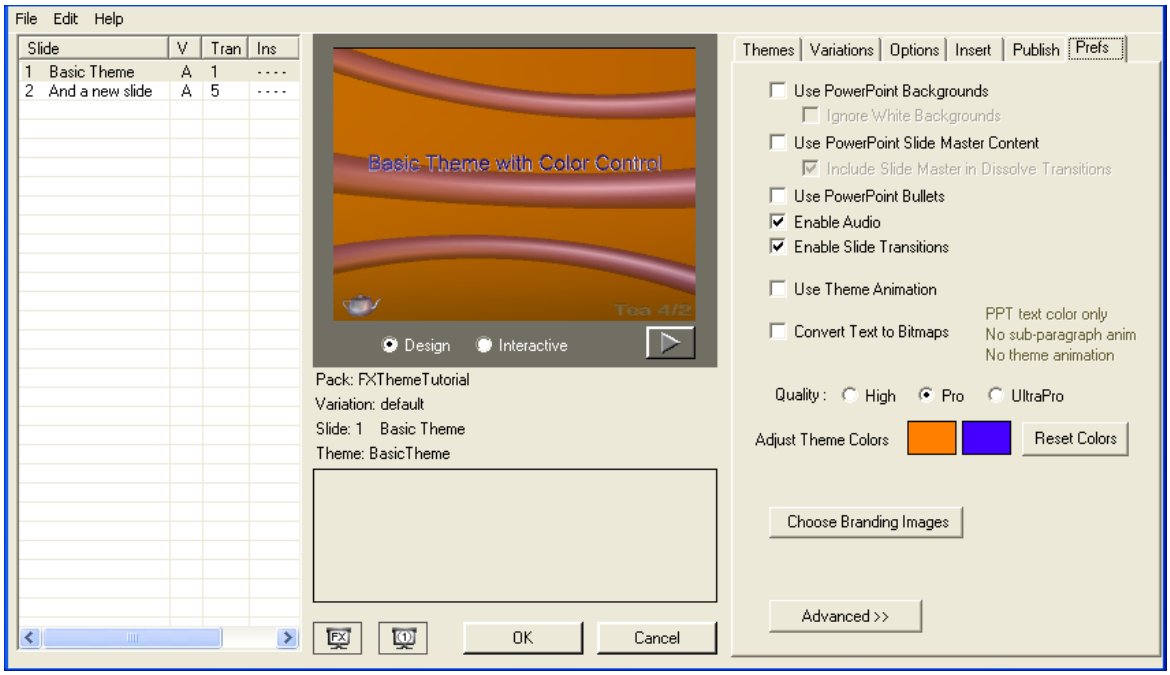


Fig 65 FXEditor with new color controllable version of the Basic Theme loaded and color adjusted.

While this particular theme does not necessarily lend itself best to user color control, the above steps illustrate the process in a way that will allow you to apply the technique when production requirements demand it.

You can also apply the same technique to creating material assets for use in coloring your PowerPoint assets. To do that, open the mtlFile.max file. Create a new box. Add a new material named something like “adjustable color swatch 1”. Then assign that box the appropriate User Defined Property. In this case that would be DiffuseColorFromSwatch=1.

Now re-export that asset file. After you press View / Refresh in FXDesigner you’ll see that you have a new material named “adjustable color swatch 1” You can assign that to any of the color attributes for your PowerPoint content such as text color. Those elements will pick up the user’s color controls.

Hint: A nice subtle touch for using user color control that does not reduce text contrast and readability is to assign it to the bevels and sides. This saves bright primary colors like white and black for text faces. Most of the FXThemes provided with Presenter use this technique.

Triggering Insert Scenes with Action Sets

Presenter is a fully interactive application. You already know that users can “interact” with both PowerPoint content and custom assets like Insert Scenes. In this section will show you how to go beyond basic Insert orientation and build Insert Scenes whose animation can be triggered in a wider variety of ways. Those will include

- Animation triggered on slide entry
- Animation triggered on slide exit
- Animation triggered by clicking on PowerPoint content.

The Presenter features that are used to implement this triggerable behavior are called “Action Sets”. A set of files that will enable you to understand both the function of Action Sets and how to create them is provided with this Authoring Toolkit. The first of those is 3ds max file that illustrates how to set up Action Sets within a Presenter Insert Scene. The second is a PowerPoint file. It shows how to set up hyperlinks in PowerPoint that trigger the Insert Scene Action Set animations. It is set up to work with the Insert Scene that is created when the provided 3ds max example file is exported.

Let’s first look at the 3ds max file. It is in the FXThemesMaxFiles folder and is named ActionSetExample.max. Open it up and you’ll see a ball, a cube, and a pyramid. The texture map for the pyramid is in the Authoring Toolkit’s Media folder.

First note that there are no User Defined Properties on any of the mesh elements. The sphere, cube, and pyramid all are rotating slowly. That rotation is accomplished by simple linear keyframes with Out of Range Controllers set to “Relative Repeat”.

Next note that there are numerous Dummy objects in the same area as the sphere, cube, and pyramid meshes. . Those all animate over 60 frames. They are 3 pairs of 2 such dummy objects. Each pair is associated by name with one of the 3 mesh shapes. You’ll see they are named as “Move Forward” or “Move Back” stand ins. As those names imply, those dummies provide the triggerable animation that will be projected onto each of the meshes in Presenter.

Looking at the User Defined Properties on those dummy objects will provide a clue as to how that is done. For example, here is the set of User Properties on the dummy object named. **SphereMoveBack_Standin**

Clip=PS

ActionSet=SphereMoveBack

ClipTarget=Sphere

These User Props say these things

Clip=PS dictates that the Position and the Scale animation on the dummy object will become the triggerable animation clips that gets used when Action Set gets initiated (triggered).

ActionSet=SphereMoveBack names the Action Set that gets created.

ClipTarget=Sphere determines which mesh will be linked to (affected by) the Action Set’s animation when it is triggered.

The other dummy objects are all similarly set up and named. Their animation takes them either from or two the pivot point of the mesh with which they are paired. Their User Properites identify what elements of their animation are to be used as triggerable clips, what mesh they are linked to, and what the name of the Action Set they create will be.

All this sets up a clear definition of Action Sets. It determines what their triggered clip animation will be, what their name will be, and what mesh they will drive. What it does not do is link them to a triggering event. That will be done in PowerPoint with a hyperlink. We’ll get to that in a moment, but first we’ll look at one more aspect of this file that has to do with two special kinds of Action Sets.

It is often the case with Insert Scenes that we look for ways to have them execute some sort of custom entry or exit from a slide. The entry is easy. You can build that into the Insert Scene and simply set the Insert to “Animate on Slide Display” in Presenter. The exit animation is something else. There’s no way to know where the Insert Scene will be in its animation cycle when the user decides to exit. Action Sets linked to Slide Exit provide a way for trigger Insert Scene animation that happens at the time a user clicks to transition to the next slide.. That SlideExit animation can be as simple and subtle as a fade or something much more dramatic.

To see how this is done, note the cubes on the left and right margins of the ActionSetExample.max file. If you scrub the timeline you’ll see that they have sequenced visibility animations. As with the dummy objects, the cubes are in pairs. There is one pair for each mesh. One is for Visibility On. The other is for Visibility Off. Look at the User Defined Properties for those control cubes. Here are the User Properties for the **vizOffCube**

Clip=V

ActionSet=SlideExit

ClipTarget=Cube

Here the User Properties mean these things.

Clip=V means that it’s the visibility of the stand in object that will be transferred as an Action Set to the target mesh.

ActionSet=SlideExit names the Action Set. In this case it’s a special reserved Action Set name that is built into Presenter. When they are encountered, these Action Sets are triggered as a slide is exited (on transition to the next slide).

ClipTarget=Cube names the mesh with which the Action Set is associated.

The obvious question would be – why not put those same visibility clips on dummy objects. The answer is that 3ds max makes it very difficult to put visibility clips on dummy objects. It's much easier to apply them to mesh objects. Since mesh objects are visible we have to locate them offscreen. We can't just turn off their "Renderable" attribute as that will prevent them from getting exported at all. We do need their User Defined Properties written into the Insert Scene's FXML file.

Once you've got a good sense for how the ActionSetExample.max file works, go ahead and export it. For the second part of this example to work correctly you should export it in the following manner. Export an Insert Scene named ActionSetExample (note – no spaces) to the Lessons Subfolder. Your FXML Exporter dialog should look like **Fig** 66.

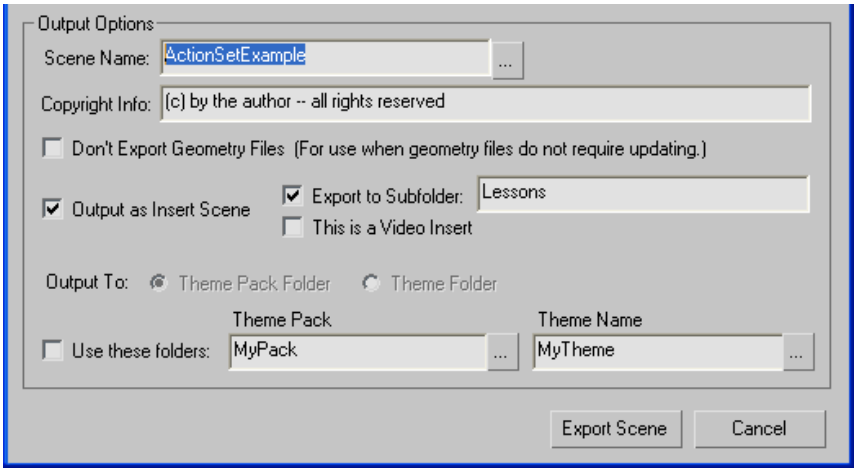


Fig. 66 shows the FXML Exporter dialog set up to export the ActionSetExample 3ds max file with naming that will enable the resulting Insert Scene to work properly in the example PowerPoint file provided with the Authoring Toolkit.

Once you have exported the ActionSetExample Insert Scene, go to the Media folder within the Authoring Toolkit and open the Action Sets Example.ppt file that you find there. In PowerPoint slide 2 of that file will look like **Fig** 67.

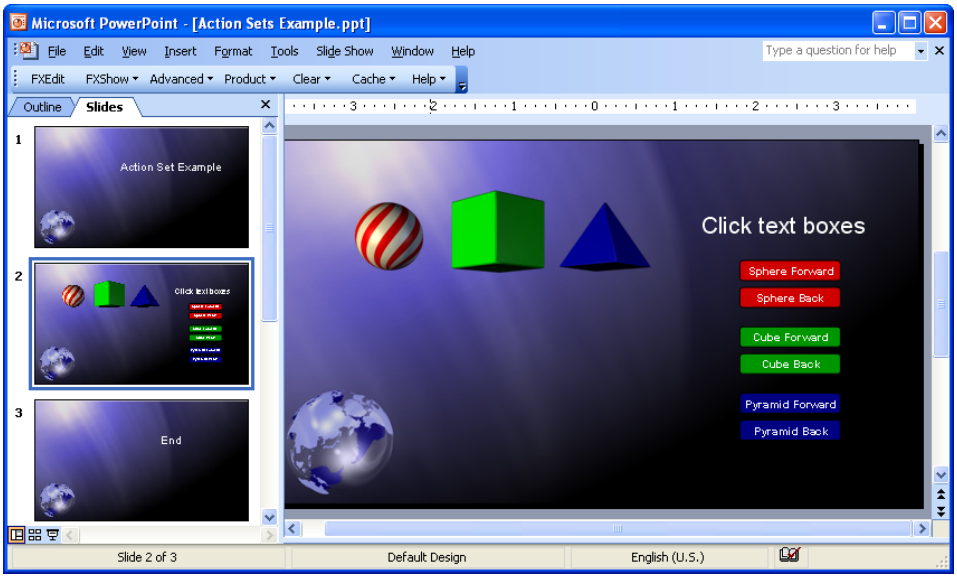


Fig. 67 shows slide 2 of the provided PowerPoint example file. It will illustrate how the triggering of Action Sets is set up using hyperlinks in PowerPoint

Start up Presenter and go to full screen display. First click from Slide 1 to Slide 2. You'll notice that the ActionSetExample Insert scene you just exported follows the SlideEnter Action Set visibility clips. The 3 meshes appear in sequence – sphere first, then cube, then pyramid. Once Slide 2 has finished drawing click on the Click text boxes. You'll see how clicks on those PowerPoint AutoShapes trigger the position and scale Action Sets. Note that anytime you click on the AutoShapes, the specified ActionSet starts immediately and from the beginning. Also note that the slow rotation of each mesh is not interrupted by the Action Set animation. It continues smoothly.

Finally press Enter or Page Down to trigger a slide transition. You'll note that the Slide Exit Action Sets all play during the transition out of slide 2.

Now press OK and head back into PowerPoint where we'll see how this is set up. Select any one of the text boxes, right click, and select the "Edit Hyperlink" option. You'll see that a Presenter specific hyperlink has been specified. For the Sphere Forward text box that hyperlink is

actionset:insert1:SphereMoveForward

This is where the triggering of the Action Set is controlled. The hyperlink editing dialog should look like **Fig** 68

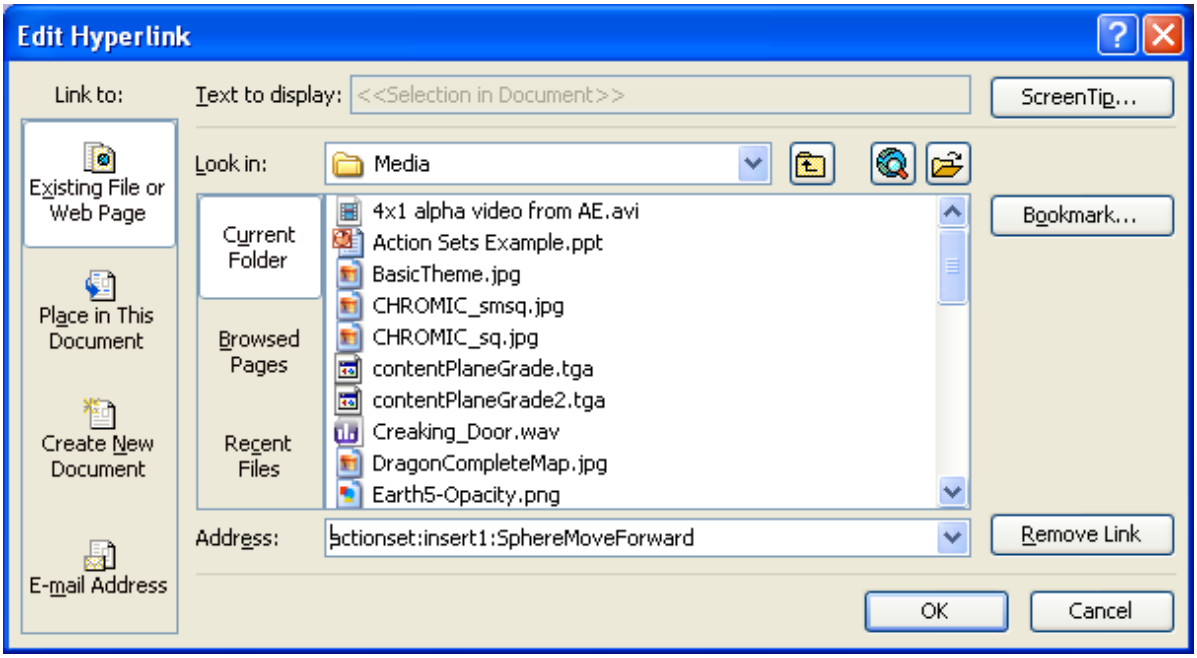


Fig. 68 PowerPoint's Edit Hyperlink dialog showing set up for triggering the SphereMoveForward Action Set on Presenter's Insert Scene 1 (vs. Insert Scene 2)

Note that the "type" of hyperlink is Existing File or Web Page but Presenter detects the leading text string of "actionset" and its Action Set logic kicks in. Also note that it's necessary to specify which of Presenter's (two) possible Insert Scene will get triggered by the named Action Set.

The SlideEnter and SlideExit Action Sets do not require this sort of hyperlink triggering as the event that kicks off their animation is known. It's the transition that triggers the exit from one slide and entry to the next.

There are several important considerations and caveats to keep in mind when using Insert Scenes with Action Sets in Presenter

- Insert Scenes that use SlideEnter and SlideExit Action Sets should always be set to "Animate Continuously" (the default setting) in Presenter. They should not be set to Animate either "On Click" or "On Slide Display" as both those setting could conflict with the Slide Enter animation.
- SlideEnter and SlideExit Action Sets should be used in pairs. If there is a SlideExit Action set in an Insert Scene, there should always be a corresponding SlideEnter – even if there is no actual animation.
- Action Set animation can be directed to the Parent of a set of objects. Done that way the animation will affect all the children. This can be a useful way to have a single animation clip (like visibility clip) affect a large number of meshes.
- Action Sets provide an additional level of control in custom 3D Presenter asset. That ability to create more dynamic entry AND EXIT Insert Scene animations can be especially useful in accentuating the stereo 3D (S3D) effect that is discussed in the next section.

Building Stereo 3D Content

With the release of S3D Presenter, it is now possible to deliver full stereo 3D (S3D) presentations. In this section we'll provide details on how to create assets that display properly in this innovative new system.

Stereo 3D is accomplished by emulating the way the human eye perceives depth. That is – a slightly different image is seen by the left eye vs. the right eye. Between those two images, objects in the foreground are "offset" horizontally relative to objects further in the background. The brain processes those two slightly different images simultaneously and generates information about the distance to objects within view. **Fig** 69 shows a simplified schematic of this concept.

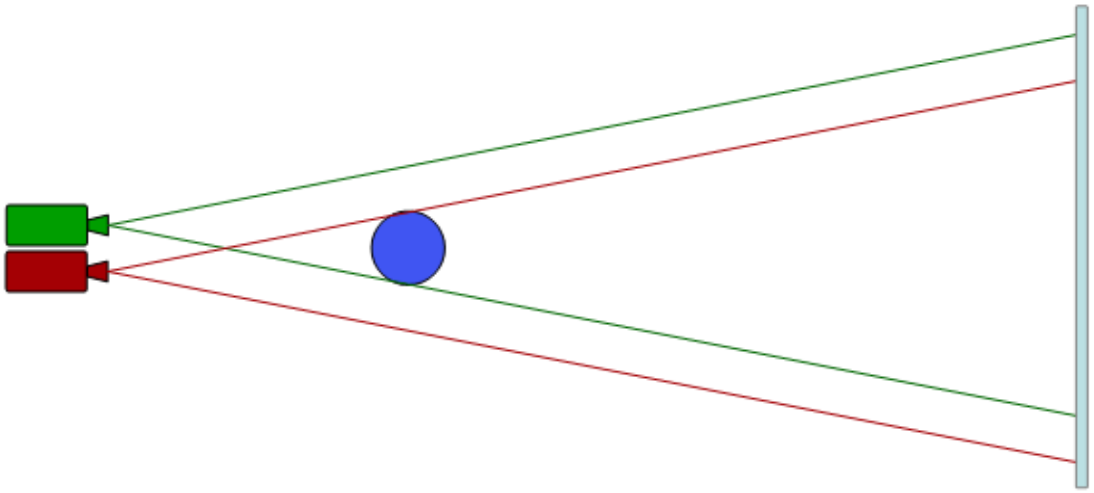


Fig. 69 Top view of a stereo viewing setup where the green camera illustrates what the left eye would see and the red camera shows what the right eye would see. Each camera's view frustum is represented with matching colored lines. Note that in the frustum of the green camera (left eye view) the blue ball in the foreground is to the far right (remember this is a top view looking down on the scene). In the frustum of the red camera (right eye view), that same foreground ball is shifted to the far left. The brain processes this different information from left and right eye and from it generates depth perception. With normal stereoscopic vision the viewer can determine how far away the blue ball actually is.

Because Presenter uses a full time 3D display architecture, creating a separate image for left and right eyes is something the system is very well configured to do. S3D Presenter has been designed to make maximum use of this advantage.

Note: Presenter also provides a number of ways to deliver final display of those two (left eye and right eye) images, but that's a different matter associated with differing S3D projection systems and monitors. For more information on those different display options and systems see the Instant Effects S3D Presenter documentation. This Authoring Toolkit discussion will focus on how to generate 3D assets that are well configured for such S3D displays.

As a starting point it's important to establish at least one definition. That is for the concept of "**Stereo Separation**". It is the distance between the two cameras in **Fig** 69 above. In a human vision setting It's also sometimes called the "interocular distance" – meaning the distance between the left and right eye. For our purposes we'll use the Stereo Separation terminology as it pertains more specifically to paired cameras viewing a 3D world. – precisely our setup.

Well define a second term for this discussion. That is "**Stereo Offset**" or just "Offset". This is the relative amount a scene element is displaced between left and right eye views. In general, objects closer to the camera have a greater Offset than objects that are further away. In the **Fig** 69 example, the dark blue ball has significantly more Offset than the light blue background plate.

To generate an asset FXAsset file that has Stereo Offset characteristics when viewed in S3D Presenter, all that is required is inclusion of a User Defined Property in the 3ds max scene from which that asset is generated. In particular, if you include this User Property

StereoSeparation=number

on the scene camera, you will generate an FXML file that generates left and right eye Offsets when viewed with proper options set in S3D Presenter. In this User Property the “number” is the distance between the two cameras. It is generally best set to about 1/60th of the distance to the plane of zero Offset (where no Offset occurs), but that may vary for different assets.

As you start working with S3D content, it is also important to remember that Presenter’s display system composites **multiple** 3D scenes together in real time. Each one of those scenes is viewed through its own camera so each of which can have its own Stereo Separation value. That allows for a very high level of control, but it can also generate results that are hard to view. In particular it is possible to generate situations where content layering looks wrong. That occurs when a scene element that has a Stereo Offset that make it look close to the viewer gets composited behind other content that has less of an Offset. There result will confuse the eye of viewers and may even cause discomfort.

For that reason it is suggested that S3D assets are built from scratch using a common viewing setup. There should be one camera and Stereo Separation value used for all 3D scenes that are created to build an S3D FXTheme. That includes foreground scenes, background scenes, transition scenes, main content scenes, and insert scenes.

Any 3D scene can be easily set up for S3D viewing simply by including the StereoSeparation User Defined Property to its camera. As such the best approach for generating S3D content is to use what you have already learned in this Toolkit and experiment. As you do that, here are some more general guidelines.

Start with a simple Insert Scene. In fact, start with the insertScene.max file that is included with the Authoring Toolkit. Open that and add a StereoSeparation=number User Defined Property. Try a value of 6 as a first experiment. Export the result, load it into Presenter, turn on the “Anaglyph” S3D viewing option as shown in **Fig** 70.

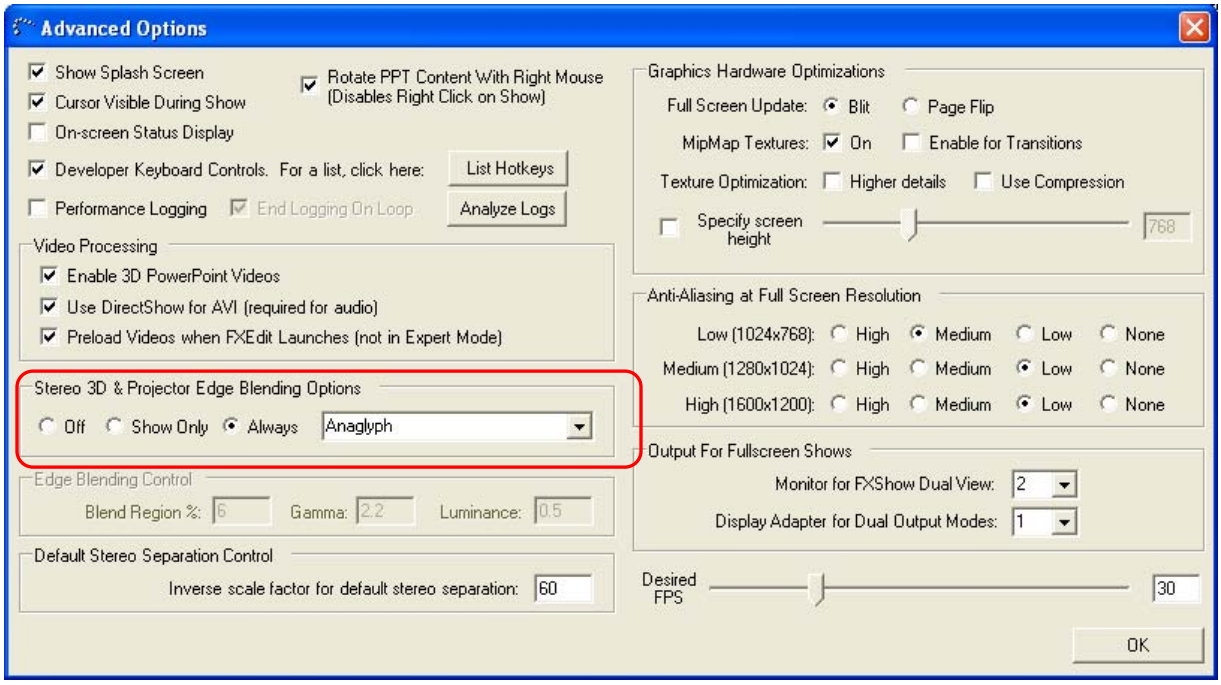


Fig. 70 shows Presenter’s Prefs / Advanced dialog with the Anaglyph Stereo 3D display option selected for display. The “Always” option means that the Anaglyph display will be used in both in full screen (Show) mode as well as in the FXPreview window within FXEdit.

Note: In order to enable S3D authoring, the Anaglyph display mode is available in all versions of Presenter. The other options that are specifically for different types of S3D monitor and projection systems are only available in S3D Presenter.

As you view the teapot in S3D mode, do so in the Basic Theme that you've created. You'll notice that it too has picked up some stereo characteristics. Specifically, the logo foreground scene sits out in front of other content. In our S3D parlance – it has Stereo Offset. That's because when you turn on any sort of S3D viewing and the FXTheme you have selected does not have specific StereoSeparation values set, Presenter assumes a value of 1/60th the distance between the scene camera and its target.

Because we've used a consistent camera setup throughout the construction of the Basic Theme, it does work reasonably well to illustrate S3D principles. When you add the teapot Insert Scene and turn on Anaglyph display you'll see something like **Fig** 71.

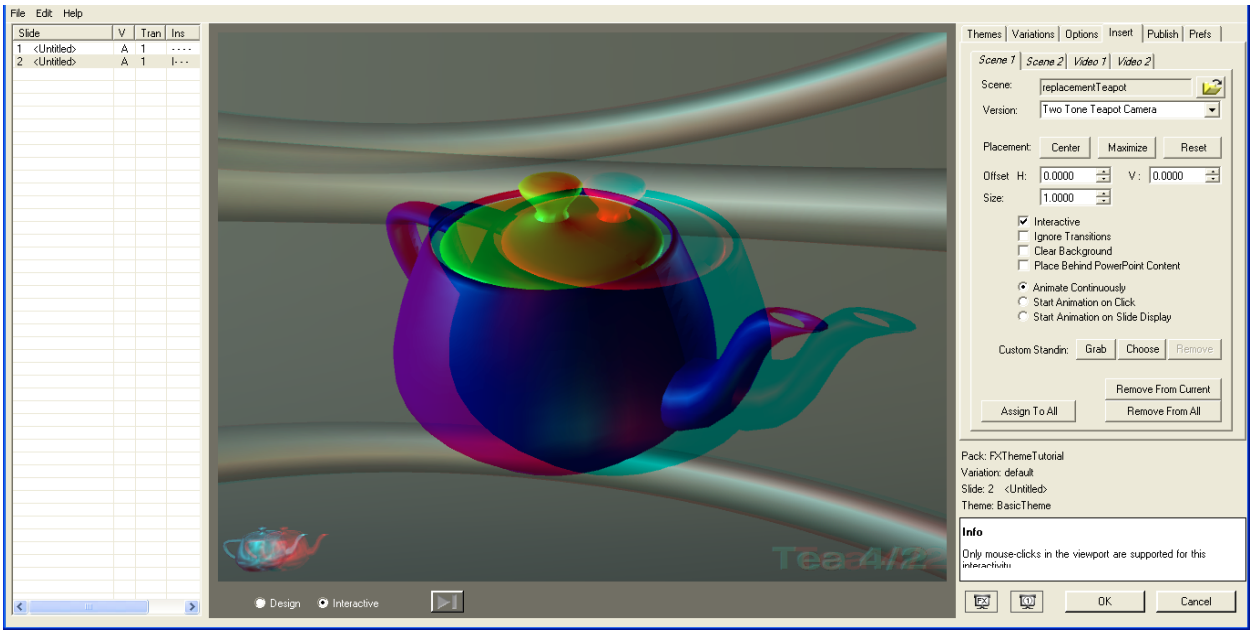


Fig. 71 shows FXEdit with Anaglyph S3D display mode set to "Always". Here the Stereo Separation for the teapot is set artificially high to accentuate the Offset. Anaglyph uses different.colors to represent left and right eye views. It is a very low tech approach to S3D that can be used with inexpensive colored glasses. It's not likely that Anaglyph display mode would be used for any professional presentation, but it does at least allow for verification that S3D mode is working.

Getting S3D assets to look great and work together well takes practice and iteration. Here are a few general tips that have been derived from discussions with customers and artists who have long experience with stereo viewing and asset creation.

- Keep the scenes low in contrast. High contrast will generate noticeable "ghosts". The amount of ghosting will vary with different S3D display technologies.
- Avoid using white in general S3D theme assets. Save that for text and / or objects, that are close to or on the plane of zero Stereo Offset. Same reason as above. White against a dark background will cause ghosting. t.
- It's OK to exaggerate depth in the scenes you create for S3D. The effect will be accentuated if you put background objects way back (and make them large). Similarly, you can shrink objects and put them much closer to the camera if you want them to pop out of the screen and be right in the viewer's face.

Naturally this will take some experimentation but the above guidelines will start you on the right path. A very rudimentary S3D example 3ds max file is provided that shows the concept of enhanced depth of field with large objects in the far field and smaller objects pulled up closer to the camera. This approach does enhance the stereo effect. The file is named S3D example file.max. It is in the FXThemeMaxFiles folder.

Packaging Sample PowerPoint files with your custom FXThemes

Presenter makes it easy for you to deliver sample PowerPoint files that show how to use your FXThemes. In the Presenter toolbar, click on Product / PowerPoint Samples. This will display the PowerPoint Sample file browser (see Fig. 72).

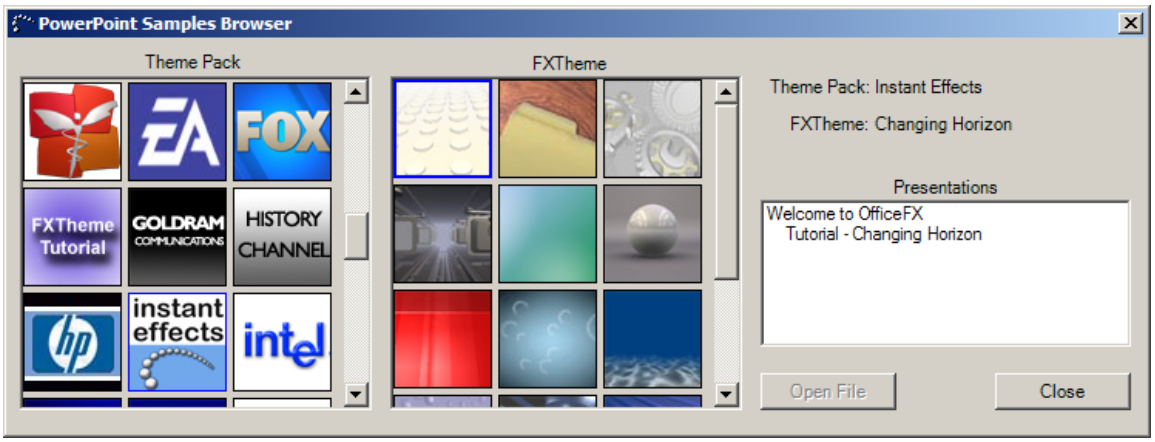


Fig. 72 – Presenter's PowerPoint Samples Browser showing selection of demo and tutorial files that are available with the standard OfficeFX 1.0 Theme Pack

Any PowerPoint files that are included within your Theme Pack or FXTheme folder will display here in the PowerPoint Samples Browser. PowerPoint files that are in the Theme Pack folder will appear fully left justified. PowerPoint files that are in the individual FXTheme folders will be slightly indented as they are in **Fig**. 72.

This feature of the Presenter user interface will allow you to provide tutorial information on how best to utilize the FXThemes you create that can be conveniently accessed by presenters who use the FXThemes you create. After you have installed the FXPackage file that is included with this Authoring Toolkit, use the PowerPoint Samples Browser to locate and run the sample PowerPoint file that uses BasicTheme.

Performance Monitoring

Everything in Presenter is done in real-time. There are many ways to create FXThemes and Insert Scenes. Some techniques that are common workflow for 3D authoring content that will be rendered offline to a sequence of image files, will yield very poor performance in Presenter. Poor performance results in choppy display which will be distracting to an audience. Some characteristics of FXML assets that can slow down playback during an Instant Effects presentation are:

- Highly detailed geometry with many hundreds of thousands of polygons. Presenter utilizes a full function 3D rendering pipeline. All those polygons have to be animated, clipped, shaded, lit, textured, and displayed by the systems graphics hardware – ideally at 30 frames per second.
- Many layers of transparency over large areas of the screen. In these cases each pixel in the transparent area will have to be rendered multiple times.
- Complex shaders. Presenter supports full DirectX 9 shaders in both FXThemes and Insert Scenes. These shaders can trigger highly complex rendering algorithms with many calculations per pixel.

In all FXTheme creation, it is important for the theme artist to understand the performance characteristics on the assets created. Presenter Version 1.2 offers a comprehensive set of tools for doing very detailed performance profiling for a particular FXTheme or Insert Scene. These tools are available in Preferences / Advanced Options (see Fig. 73). In the panel that appears turn on Performance Logging and the system will record the frame rate and number of skipped frames for each slide and transition in a PowerPoint presentation.

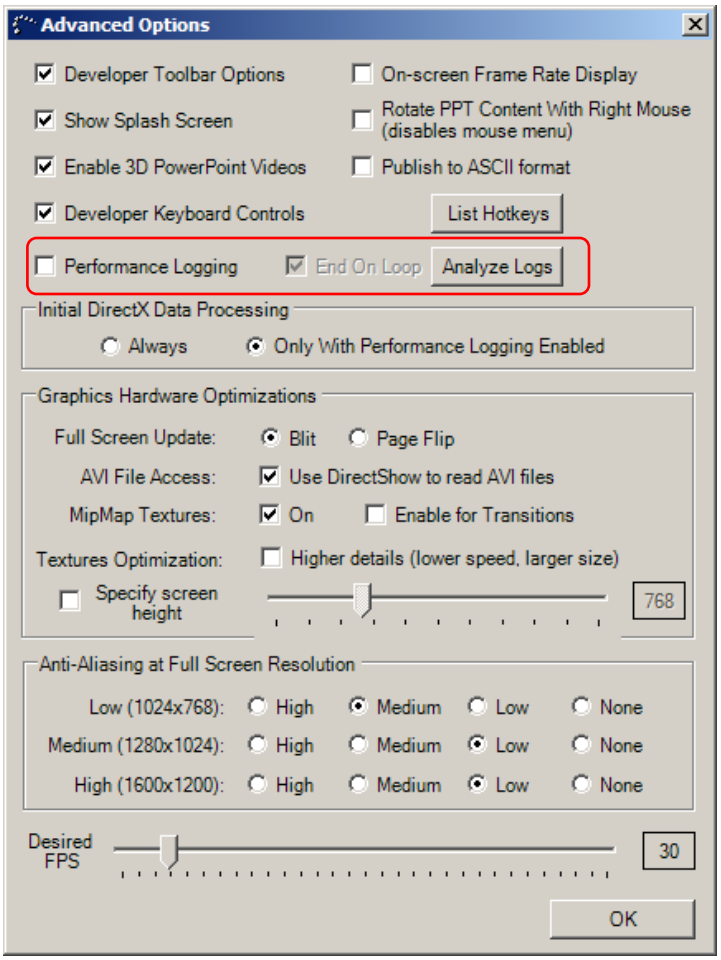


Fig. 73 – Presenter Version 1.2 Preferences / Advanced Options panel with Performance Logging options highlighted in red.

For best repeatable testing results, enable Automatic Advance in your PowerPoint file from PowerPoint's Slide Show / Slide Transitions interface. Enter the advance time in seconds. Also go to the Slide Show / Set Up Show panel and click on "Browsed at Kiosk". This will cause the presentation to cycle continuously.

Now in Presenter go to Preferences / Advanced Options and check the End On Loop box next to Performance Logging. Finally, go into full screen playback of your test file. With Automatic Advance enabled, it will play by itself. With the End On Loop option checked it will only play once, and then drop back out to the FXEditor. Display speed characteristics for each slide and transition will be logged. Clicking on Analyze Logs will allow you to select the log file for inspection. That file will be stored under the name of the PowerPoint file that was used to create it. The browser will take you to the folder where performance analysis logs are stored. The Analyze Logs utility will consolidate results into an easily interpretable form. See Fig. 74.

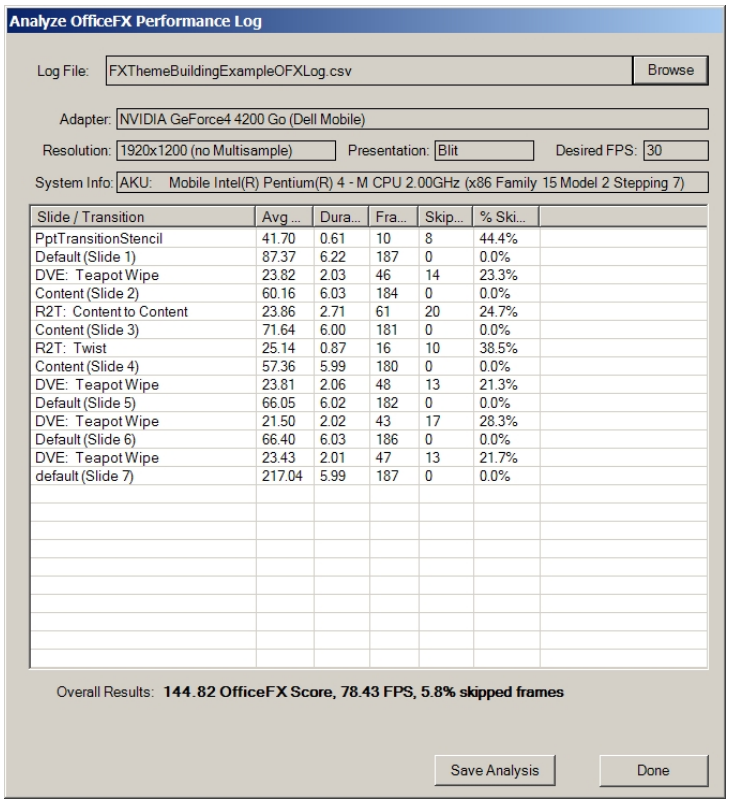


Fig. 74 Performance log file for the BasicTheme using the Sample PowerPoint file provided with the Theme Authoring Toolkit. Raw log files are not in this convenient viewing and summary analysis form. The Save Analysis button on the Performance Log user interface allows you to save log files in this form. Saved analysis files are in Excel spreadsheets.

Presenter is a real-time application, but it is not precisely like a computer game in the way it displays. By default it tries to display at 30 frames per second. In the Advanced Options dialog, the user can reset that target frame rate. In general it should be left alone, but in certain circumstances a presentation will perform more smoothly if the target frame rate is actually set lower than 30 fps. In contrast, computer games typically run in a mode where they constantly try to achieve the highest possible frame rate. As a result, most 3D applications and hardware are benchmarked on the basis of frames per second.

Frame rate is also important for Instant Effects presentations, but more important is the perceived "smoothness" of the display. Presenter is constantly changing its graphics context. As a presenter moves from slide to slide, different FXTheme variations are employed which can significantly change the nature of how the user's information is displayed. Transitions add another layer to these graphics context changes. They may swap in entirely different scenes and assets.

As a result, while hypothetical maximum frames per second is important, for Presenter a more important variable is percentage of frames skipped. The higher this percentage of skipped frames, the jerkier the playback will appear in Presenter.

The Presenter Performance Log reports both hypothetical maximum playback frame rate achieved for each slide and transition. It also records the actual frame rate and the number of frames skipped. The Presenter Score reported is a function of both average frames per second achieved and the number of frames skipped.

For FXTheme development it is a good idea to use the Presenter Performance Logging tools to assess the performance of the assets you create. These tools will allow you to make precise comparisons as you look for ways to optimize your work.

Appendix 1 – Presenter's User Defined Properties

User Defined Property	Recipient	Function
Clip=M	Mesh	Material animation on the mesh is becomes a triggerable Clip
Clip=V	Mesh	Visibility animation on the mesh is becomes a triggerable Clip
Clip=P	Mesh	Position animation on the mesh is becomes a triggerable Clip
Clip=R	Mesh	Rotation animation on the mesh is becomes a triggerable Clip
Clip=S	Mesh	Scale animation on the mesh is becomes a triggerable Clip
Clip=PRV*	Mesh	Position, Rotation, and Visibility animations on the mesh are used in Clips
ShaderIncludeFile=shadername.xml	Mesh	Links a user provided shader to the mesh
SIF=shadername.xml	Mesh	Links a user provided shader to the mesh
R2TMaterial=CurrentSlide	Mesh	Texture maps image of current slide to the R2T mesh
R2TMaterial=CurrentSlideWithOpacity	Mesh	Texture maps image of current slide to the R2T mesh. Includes any Opacity animation
R2TMaterial=NextSlide	Mesh	Texture maps image of next slide to the R2T mesh
R2TMaterial=NextSlideWithOpacity	Mesh	Texture maps image of next slide to the R2T mesh. Includes any Opacity animation
VideoStandin=true	Mesh	Designates mesh that will receive user specified video in a Video Insert Scene
OpacityReferencesDiffuse=true	Mesh	For Video Inserts that will take video with alpha (transparency)
Deformation=Loop	Mesh	Exports deformations as vertex cache (vtx) files. Animation loops continuously
Deformation=Constant	Mesh	Exports deformations as vertex cache (vtx) files. Animation plays once.
Deformation=PingPong	Mesh	Exports deformations as vertex cache (vtx) files. Animation plays continuously forward then back.
DeformationFPS=number	Mesh	Specifies frame rate for vertex cache sampling in frames per second
DeformationTransClip=true	Mesh	Specifies that a deformation will be used as a clip in transition.
ZOffset=number**	Mesh	Pushes mesh specified number of units away from camera for Z sorting

DiffuseColorFromSwatch=number	Mesh	The mesh will take its diffuse material property from the numbered user input control swatch (possible numbers 1-4)
ActionSet=name	Dummy	Creates an Action Set of the specified name
ClipTarget=mesh name	Dummy	Specifies the mesh which will be affected by the dummy's animation clip
StereoSeparation=number	Camera	Specifies Stereo Separation
InteractionDely=number	Camera	Number in seconds that an Insert Scene will delay before allowing user interaction
ResetView=true	Camera	Resets an Insert Scene to its originally designed orientation any time the slide it is on is accessed.

* Clip User Properties can be combined. Order is not important. Clip=PRV is equivalent to Clip=VRP

** replace number with some integer in units used in your 3ds max file

*** numbers from 1 to 4