Cisco Digital Signage

Content Creation Best Practices Guide

For Digital Media System

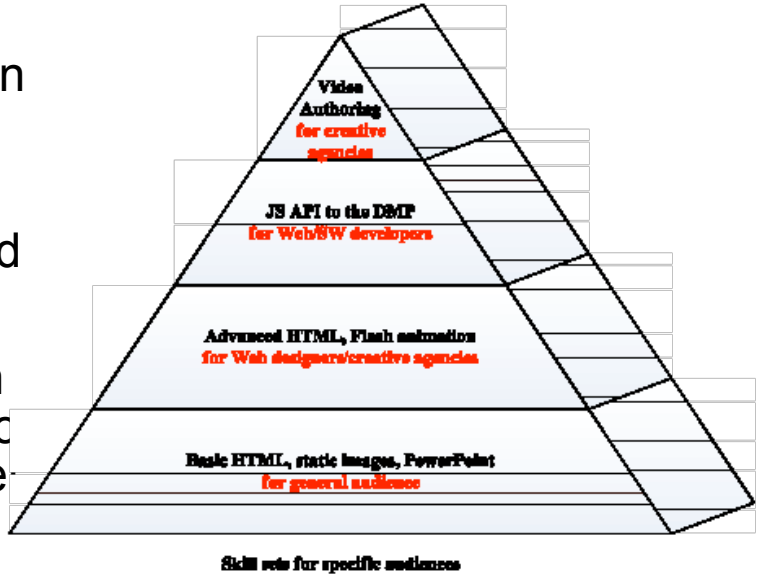
August 1, 2008

Purpose

- The purpose of this presentation is to educate media professionals, vendors, creative agencies, interested parties, etc. on how to create optimized content for the Cisco Digital Media System (DMS) for Digital Signage.
- This guide will help customers to adopt best design practices and create content suited to Cisco Digital signage playback. Content may range from video clips, Flash animations, static images, HTML pages, PPT presentations, or a combination thereof and displayed in either full-screen or within particular screen zones.
- JavaScript applications can be developed to enable various functionality (dynamic content polling, event triggers to change signage content, etc.). Video and graphics can be interlaced.

Intended Audience and Recommended Tools

- Anyone who has basic HTML and web design knowledge can create signage content.
- It is not a given, however, that content that looks good on a computer will also look good on a signage display.
- Also there are various tools that the user can adopt to create appropriate content – refer to the adjacent pyramid to determine what level of content complexity is appropriate for you.
- The tested and recommended tools for content creation include the following:
  - Adobe Dreamweaver
  - Adobe Flash
  - Adobe Photoshop
  - Adobe Premiere
  - Adobe After Effects
  - Final Cut Pro
  - Sony Vegas
  - VLC



Content Creation Pyramid

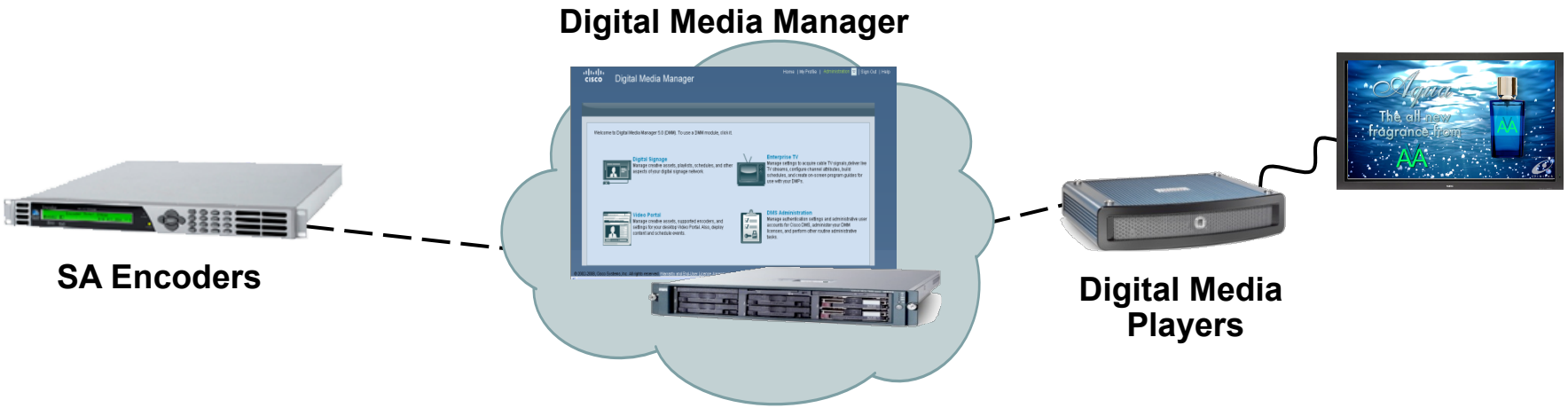
Table of Contents

Digital Signage and Digital Media System Overview

Return to Contents

Cisco Digital Signage: Overview

A comprehensive solution for flexible and centralized management and publishing of digital media to networked, on-premise digital signage displays



Cisco Digital Media Manager for Digital Signage
Centralized Digital Media Management and Publishing

- Web-based media management and publishing
- Granular access control for roles such as: designers, IT admins, marketing/ sales, regional vs. corporate
  - Sales and Marketing – create playlists, scheduling and daypart
  - Content creators/designers – manage assets, design screen layout and zones
  - IT admins – can configure, remotely manage, group and run reports on DMPs
  - Control display properties of the display via connected DMP
- Integrates with Cisco ACNS for optimized WAN delivery

Same Application for Managing and Publishing Cisco Desktop Video Content

Cisco Digital Media Players
Playback of Compelling Digital Media Content

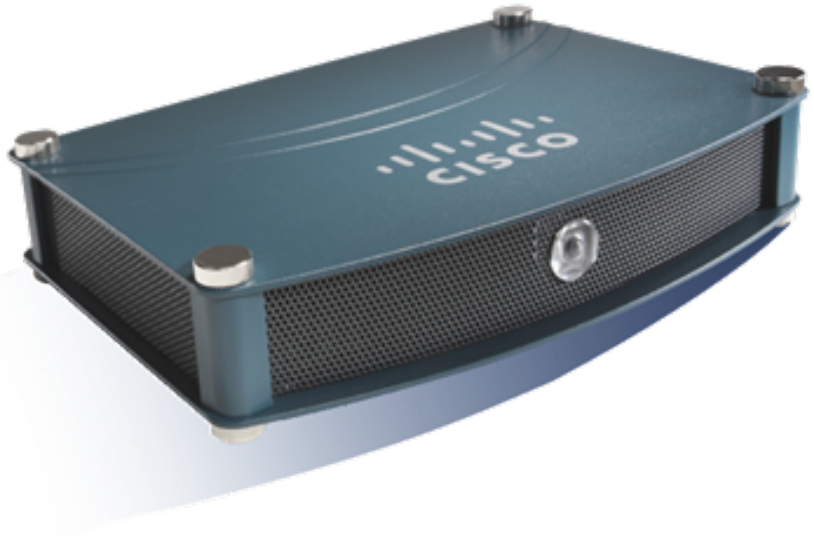
- Renders graphics, Web content, text tickers on digital displays.
- Supports full-screen or “zoned” video in SD or HD resolutions.
- Customizable on-screen templates.
- Remote management of display properties.
- IP-network addressable.
- Local storage, high availability, automatic failover.
- Small and lightweight.
- Remote control for interactivity.
- Low power consumption and high reliability.
- Security: Hardened device.

DMP4305G

DMP4400G

Cisco Digital Media Player 4305G

- Supports: MPEG 1, 2, and 4 Part 2 in standard definition (SD) and HD, graphics, Web content, Adobe Flash 7 and earlier animation, and tickers.
- Small and lightweight: 7.5” x 5” x 1.5” at 1 lb.
- Local storage of 2-GB capacity.



Cisco Digital Media Player 4400G

- Supports: MPEG 1, 2, and 4 part 10 in standard definition (SD) and HD, graphics, Web content, Adobe Flash 9 animation, and tickers.
- Small and lightweight: 10” x 8” x 2” at 4.5 lb.
- Local storage of 4-GB capacity.

Digital Signage Content

How is Signage Content Created?

	Source	Creation tools for On-Demand Playback	Creation tools for Live Playback
Flash Animation	• Graphics, photos, and images • PPT • HTML • Other flash animations (.swf)	• Adobe Flash 6 +	NA
HTML, Web Content	• HTML, web pages • Excel spreadsheets • RSS and Text data	• Web editors (WYSIWIG): Dreamweaver, Front Page, etc. • Excel (output to HTML)	NA
Video: MPEG1 MPEG2 MPEG4 (H.264)	• DVD (MPEG-2) • Analog formats:* (beta tape, DV-Cam) • Other digital formats: .avi, .mov, .flv, .mpg)	• Adobe Premier Pro • Apple Final Cut Pro • VLC • Sony Vegas	NA
Video: MPEG-2	• Live camera • Live cable feed	NA	• Scientific-Atlanta D9022/D9032/D9034/ D9034-S/D9054 Encoders**

* Analog formats require a beta tape or DV-Cam reader deck

** Cisco Digital Media Encoders are **NOT** currently compatible with Digital Media Players

Content Management Options

There are two ways to publish content for playback on the Digital Media Player:

1. **Using the Digital Media Manager (DMM)**

   This allows you to customize pre-defined templates, select content, and publish the content changes to the DMP on a scheduled or ad-hoc basis.

2. **Stand-alone creation and publishing of standalone web content that plays on the Digital Media Player (DMP)**

   Create your own web pages and templates, and publish content using your own tools and publishing processes. Then use the DMM or the DMP's Device Manager interface to instruct the DMP to play that content back by fetch a URL or subscribibing to a multicast.

What's the difference?

For standalone content, the DMM cannot be used to make changes to the content. The DMM can only be used to instruct the DMP to locate the content by URL for content playback.

You are not required to use the DMM to create templates though emplates created outside of the DMM must be self -managed.

Content Formats

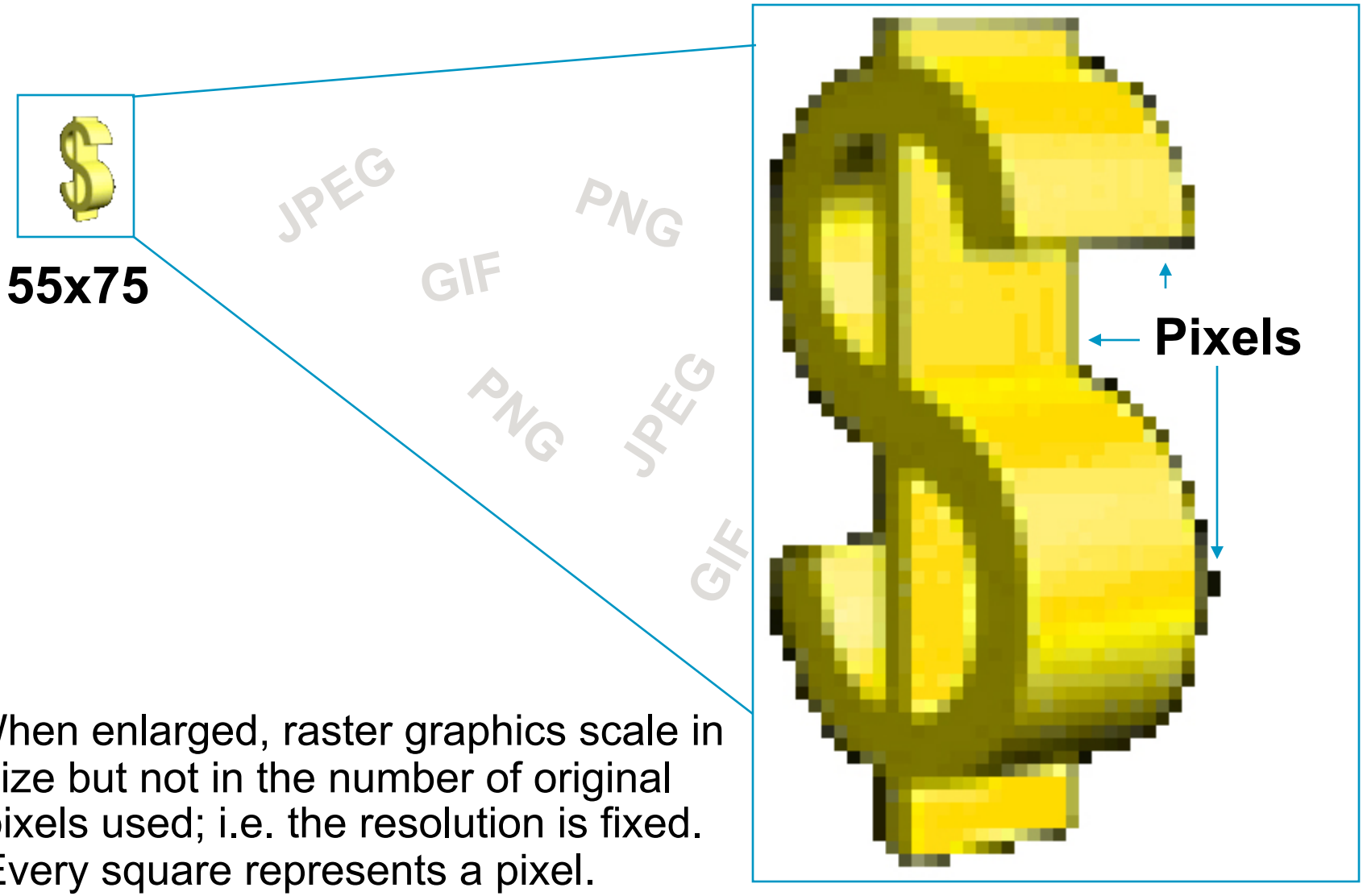
Return to Contents

Acceptable Content Formats for DMP

DMD	Custom	
Yes	Yes	▪ ***Flash10*** compatible with down to ***Flash 6*** movie (.swf); flash with no audio support
Yes	Yes	▪ ECMA script (***Java Script***) version ***1.7***
Yes	Yes	▪ ***Mozilla 2.012***
Yes	Yes	▪ File formats for the animated, or still images: ***.swf*** , ***.gif, . jpg,_.png***
Yes	Yes	▪ ***MPEG-2*** encapsulated in ***Transport Stream***, High Definition (***HD***) and Standard Definition (***SD***) formats, and size depending on native resolution of your display.
Yes	Yes	▪ Supported Bit rate up to ***15 Mbit/sec*** recommended for : HD: 12 Mbit/sec - 15 Mbit/sec SD: 3 Mbit/sec - 5 Mbit/sec
Yes	Yes	▪ ***h.264 in MPEG-2 container*** recommended for : HD: Approximately 6 Mbit (Min3 Mbit) SD: Approximately 5 Mbit (Min2 Mbit)
Yes	Yes	▪ ***RSS*** tickers via RSS XML

Supported Graphics Files Formats

Supported Graphic File Formats: Raster Graphics



When enlarged, raster graphics scale in size but not in the number of original pixels used; i.e. the resolution is fixed. Every square represents a pixel.

Supported Graphics Files Formats: Raster versus vector graphics

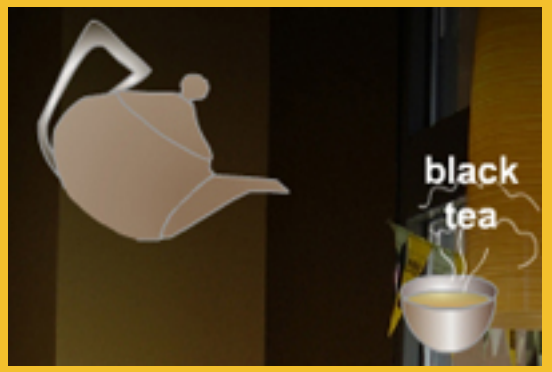
Raster Graphics:

- GIF 
- JPEG 
- PNG 

***Fixed resolution** (including video)*

Vector-based Graphics:

- **SWF** (no embedded video)



Scalable resolution

Supported Video Format MPEG-2

- *720p* is widely used and supported by most plasma and LCD displays. For example, 720p is used by *ABC* and *ESPN* because the smoother image is desirable for fast-action sports telecasts.
- The DMP will play all standard and high definition formats which your LCD or plasma display supports.
- Stream type: transport stream.

Transport stream

Allows multiplexing of digital video and audio and to synchronize the

Supported Fonts

Return to Contents

Supported Fonts

- ***Albany regular - similar to Arial***
  - all Central+East+West European languages.
- ***Verdana regular- same as Windows Verdana***
  - all Central+East+West European languages.
- ***Traditional Chinese.***
- ***Simplified Chinese.***
- ***Japanese (Square Gothic).***
- ***Arabic (Universal Off The Shelf).***

Note: when a font is embedded within a Flash file the font will display correctly even if the corresponding font is not installed on your DMP.

Supported True Type Fonts

These TrueType fonts are preinstalled as part of this release:

Name	Filename	Typographic Sample
Vera Sans	Vera.ttf	ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefgh ijklmnopqrstuvwxyz1234567890!@#$%^&
Vera Sans Bold	VeraBd.ttf	**ABCDEFGHIJKLMNOPQRSTUVWXYZabc defghijklmnopqrstuvwxyz1234567890**
Vera Sans Bold Oblique	VeraBI.ttf	***ABCDEFGHIJKLMNOPQRSTUVWXYZabc defghijklmnopqrstuvwxyz1234567890***
Vera Sans Oblique	VeraIt.ttf	*ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefgh ijklmnopqrstuvwxyz1234567890!@#$%^&*
Vera Sans Mono	VeraMono.ttf	ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghij klmnopqrstuvwxyz1234567890!@#$%^&*()
Vera Sans Mono Bold	VeraMoBd.ttf	**ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghij klmnopqrstuvwxyz1234567890!@#$%^&*()**
Vera Sans Mono Bold Oblique	VeraMoBI.ttf	***ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghij klmnopqrstuvwxyz1234567890!@#$%^&*()***
Vera Sans Mono Oblique	VeraMoIt.ttf	*ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghij klmnopqrstuvwxyz1234567890!@#$%^&*()*
Vera Serif	VeraSe.ttf	ABCDEFGHIJKLMNOPQRSTUVWXYZabcd efghijklmnopqrstuvwxyz1234567890!@#$
Vera Serif Bold	VeraSeBd.ttf	**ABCDEFGHIJKLMNOPQRSTUVWXYZab cdefghijklmnopqrstuvwxyz1234567890**

Supported Bitmap Fonts

These X11 bitmap fonts are preinstalled as part of this release:

Foundry	Family Name	Weight Name	Slant	Setwidth Name	Add Style Name	Pixel Size	Point Size	Resolution X	Resolution Y	Spacing	Average Width	Charset Registry	Charset Encoding
adobe-	helvetica-	bold-	r-	normal-	-	0-	0-	75-	75-	p-	0-	iso8859-	1
adobe-	helvetica-	bold-	r-	normal-	-	12-	120-	75-	75-	p-	70-	iso8859-	1
adobe-	helvetica-	bold-	r-	normal-	-	14-	140-	75-	75-	p-	82-	iso8859-	1
adobe-	helvetica-	bold-	r-	normal-	-	18-	180-	75-	75-	p-	103-	iso8859-	1
adobe-	helvetica-	bold-	r-	normal-	-	24-	240-	75-	75-	p-	138-	iso8859-	1
b&h-	lucida-	bold-	l-	normal-	sans-	0-	0-	75-	75-	p-	0-	iso8859-	1
b&h-	lucida-	bold-	l-	normal-	sans-	12-	120-	75-	75-	p-	79-	iso8859-	1
b&h-	lucida-	bold-	l-	normal-	sans-	14-	140-	75-	75-	p-	92-	iso8859-	1
b&h-	lucida-	bold-	l-	normal-	sans-	18-	180-	75-	75-	p-	120-	iso8859-	1
b&h-	lucida-	bold-	l-	normal-	sans-	24-	240-	75-	75-	p-	152-	iso8859-	1
misc-	fixed-	medium-	r-	normal-	-	7-	50-	100-	100-	c-	50-	iso8859-	1
misc-	fixed-	medium-	r-	normal-	-	7-	70-	75-	75-	c-	50-	iso8859-	1
misc-	fixed-	medium-	r-	normal-	-	8-	60-	100-	100-	c-	50-	iso8859-	1
misc-	fixed-	medium-	r-	normal-	-	8-	80-	75-	75-	c-	50-	iso646.1991-	irv
misc-	fixed-	medium-	r-	normal-	-	8-	80-	75-	75-	c-	50-	iso8859-	1

- 5x7
- 5x8
- 6x13
- cursor
- fixed

DMM Capabilities

Return to Contents

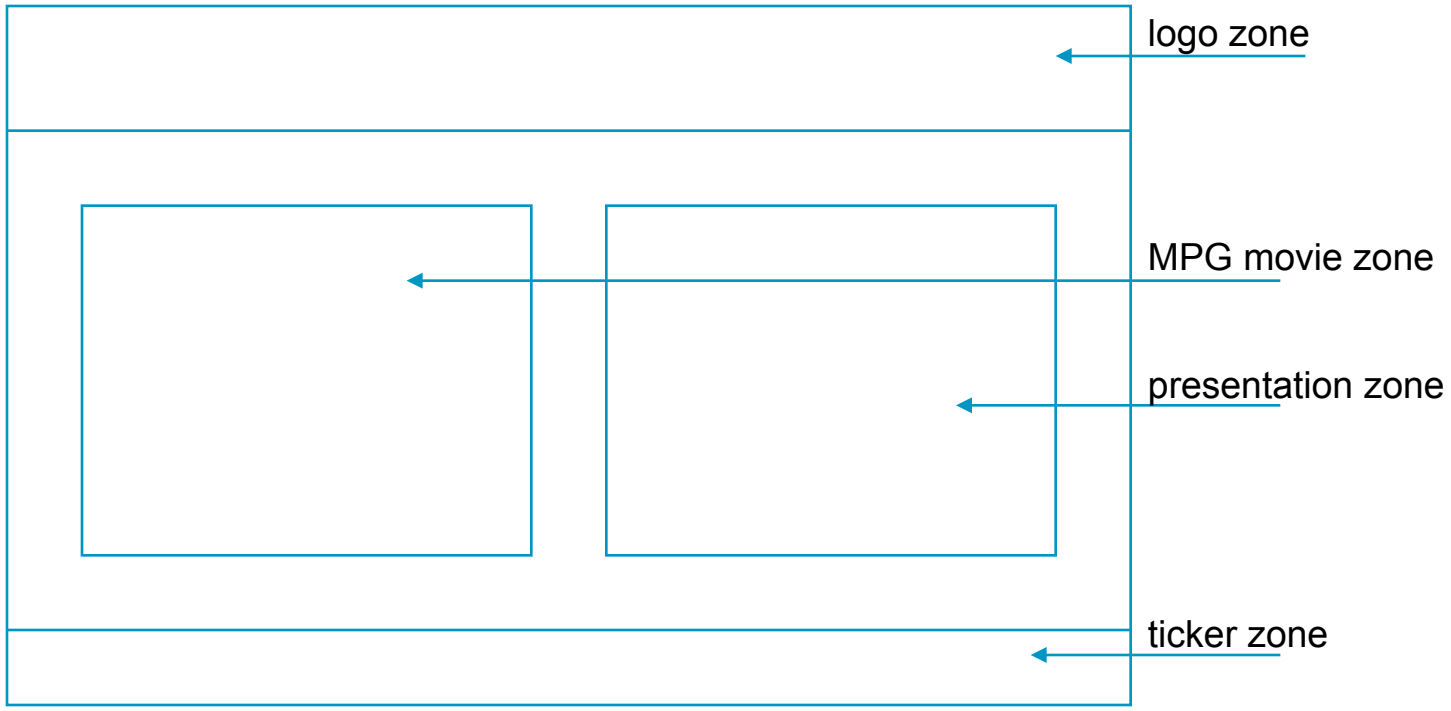
DMM Capabilities

- You can upload, manage, and compose on-screen designs.
- For on-screen design creation the DMM allows you to utilize **Digital Media Designer** tool. **Digital Media Designer** helps you to subdivide the screen space into ***rectangular*** areas called ***screen zones*****, and** to select the media objects to play back in those screen zones – including media from ***digital video*** files, ***image*** files supported by **Mozilla Firefox** browsers, ***RSS Feeds,*** ordinar***y web pages*** and other media files.
- You can upload, create, and manage playlists and publish content that will run on your DMP at scheduled or ad hoc times.
- Playlists consist of MPEG and SWF files that can play in one or two zones of the screen (**or full-screen**). You may also edit a playlist after it's creation.
- In the **Digital Media Designer,** you may choose from a selection of 6 pre-defined, modifiable, templates/layouts or upload your own designed media.
- User Guide for Cisco Digital Media Manager 5.0 - Overview [Cisco Digital Media Manager] - Cisco Systems

DMM Interface

The Zones Definition

- ***4 zones template example***: logo zone, movie zone, presentation zone, and ticker zone:



DMM Designer:
Benefits

- For ***standalone demo*** all of the multimedia content including **flash animations**, **java script**, **mpg2** movies is included into a html page and plays in **Mozilla Firefox browser**. Every object on the page has absolute positioning in tag on html page.
- Using **DMM** you can **schedule content** to be published using the playlist scheduler.
- For both ***standalone demo and for DMM demo*** you can upload and run content from the server or SD card. Demo files' bandwidth should not exceed Max capacity of your USB storage (up to almost 2GB if you have 2GB USB).
- In a ***standalone demo*** you have to create content manually. The **DMM Designer** has pre-designed templates that can save you time in template/layout creation.

Logo area

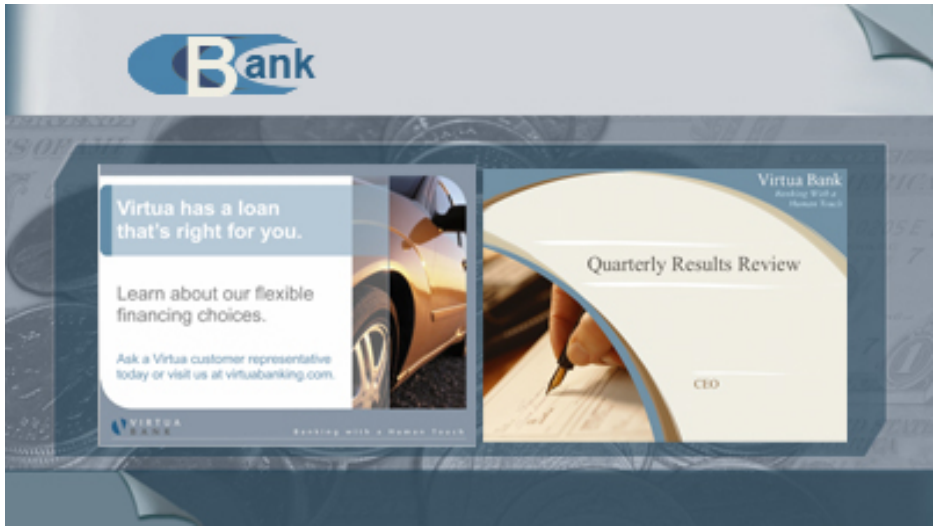
MPG2 movie

png, jpg, gif image, or swf

Ticker

DMM Designer:
Examples of Customized Templates

- These templates were customized and managed using DMM.
- You can place your logo, static or dynamic RSS tickers, MPEG and flash presentation.
- You can publish these templates and their related content to the DMP for local playback or to a network server for streaming.



Presentation Creation in the Digital Media Designer

Getting Started

Return to Contents

DMM Designer:
Prerequisites

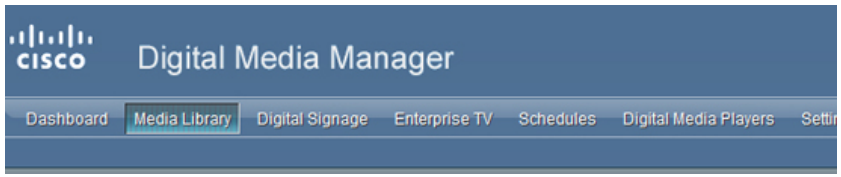
- 1) Ensure that **Java Runtime Engine** (JRE) 1.6.0_01 or higher is installed on your machine and is working correctly.
  For more information please read User Guide for Cisco Digital Media Manager 5.0
- 2) Select **Start** > **Setting** > **Control Panel** > **Internet Options**, then click the **Advanced** tab. Scroll the browsing area, deselect the **Enable third party browser extensions** (**requires restart**) check box, then click **OK.**
- 3) Ensure that your DMPs use **firmware version 1.01-RC18 or later**. You cannot use any earlier firmware version. *To check the firmware version on a DMP, log in to each DMP's Device Manager interface, then click* **Hardware and Firmware Versions.** If you need a firmware update, go to:

http://www.cisco.com/cgi-bin/tablebuild.pl/dms.

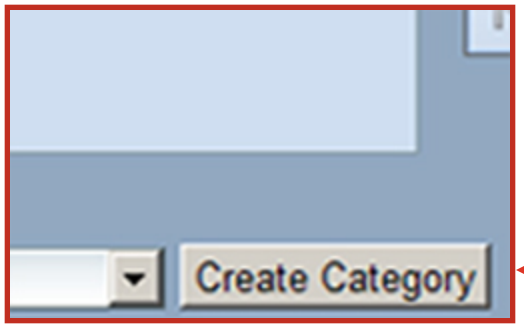
On-Screen Presentation Creation
Uploading Content

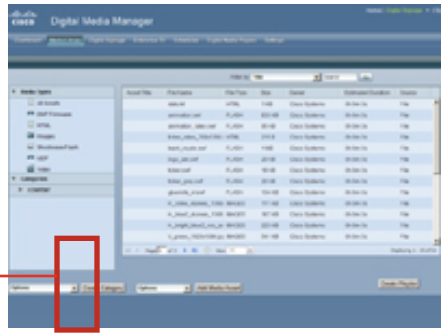
- *Step1* : ***Select the*** **Digital Signage Module** icon **Digital Media Manager** page.

- *Step2* : ***Click*** on **Media Library** tab to open the content library.



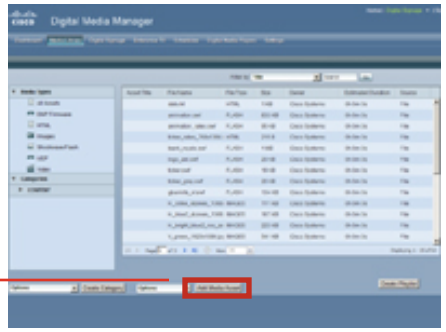
- *Step3* : Press the "**Create Category**" button to create new category folder.



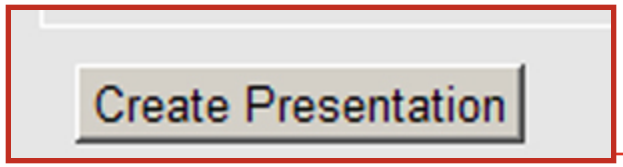


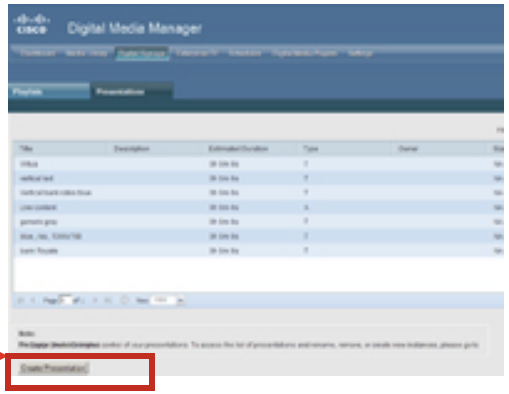
On-Screen Presentation Creation
Launching the Designer

- *Step4* : ***Click*** on the "**New**" icon to upload new content.



- *Step5* : **Under** the **Digital Signage** tab open the **Presentations** tab
- *Step6* : ***Click*** on **Create Presentation** button to create your presentation.

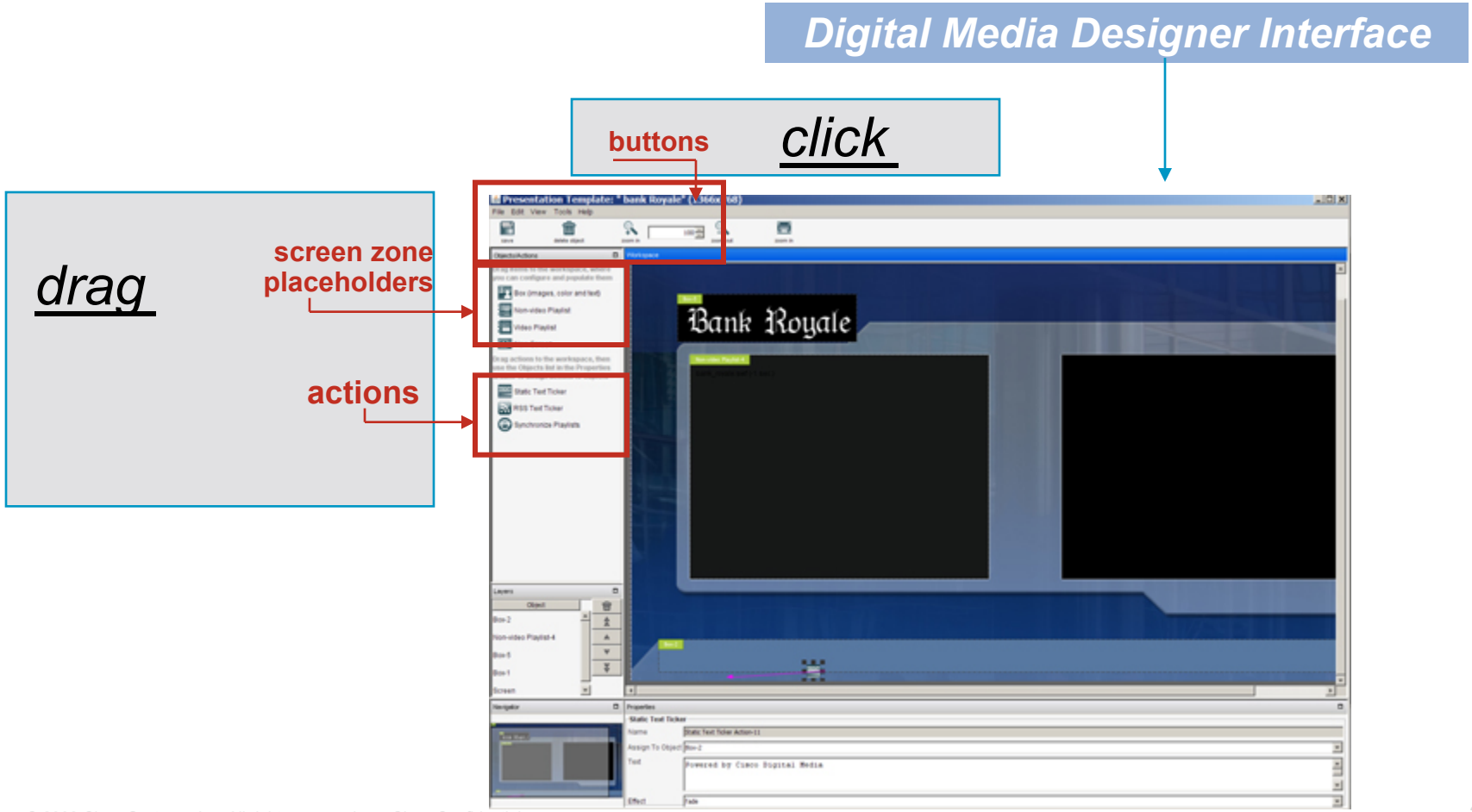




Note: you can skip Steps 2 and 3 if you are not going to upload media files for your on-screen design

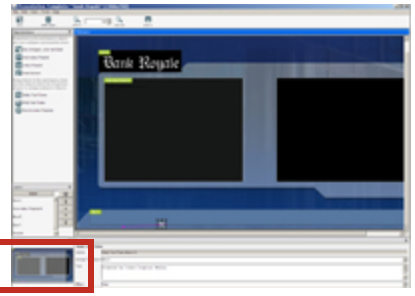
On-Screen Presentation Creation
Understanding the Designer Toolbar

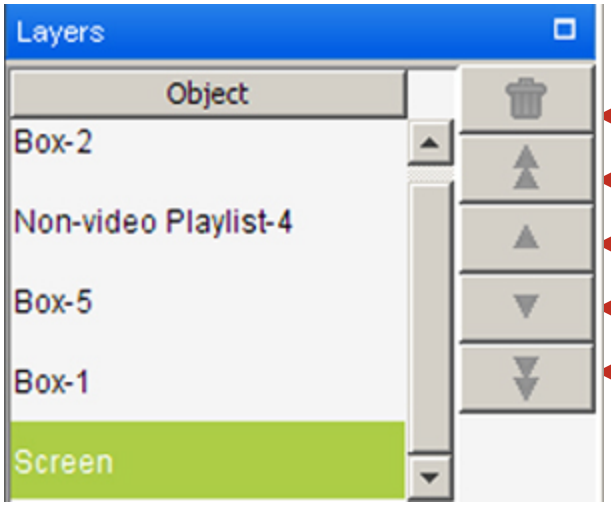
- The toolbar in Digital Media Designer contains **buttons** that you ***click***, **screen zone placeholders** that you ***drag*** to the workspace, **actions** that you ***drag*** to the workspace (where you associate them with screen zones), and controls for the degree of workspace magnification.



On-Screen Presentation Creation
Understanding Layers

- The Designer ***creates*** a **new** **layer** every time you ***drag*** an **object** to the workspace. Objects are sorted by their placement on different layers. Each layer holds only one object, with each object representing one screen zone, and is a top-down hierarchical display layout. Use the buttons in the Objects panel to order the selected object, as follows:





Deletes the selected screen zone object.

Moves the selected screen zone object to the absolute top layer.

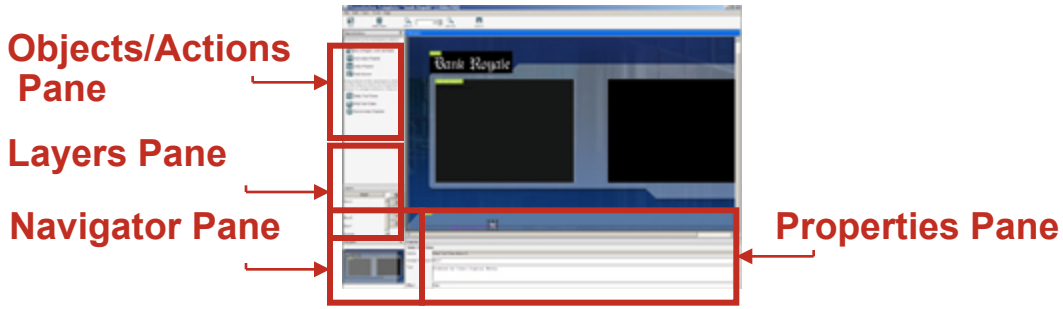
Moves the selected screen zone object up by one layer per click.

Moves the selected screen zone object down by one layer per click.

Moves the selected screen zone object to the absolute bottom layer.

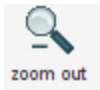
On-Screen Presentation Creation
Tool Panes

- **Panes** in the Digital Media Designer may be ***moved***, ***resized***, ***expanded***, or ***collapsed***, and contain features to help you design layouts for signage. The **panes** are labeled: **Layers**, **Objects/Actions**, **Navigator**, and **Properties**.
- All panes are open by default.



Notes:
- *To move a pane, drag it by it's title bar. The only constraint is when you move a pane, one side of it will always be pinned to the workspace.*
- *To change the width or height of a pane, click and drag a panel's border.*
- *To collapse a pane, click the square button at the right end of its title bar. In its collapsed form, the entirety of a pane is reduced to button form, where the button shows nothing except the pane name in very small type and is pinned to one edge of the Digital Media Designer window.*
- *To reopen a pane that you collapsed, click where you see its name pinned to the edge of the Digital Media Designer window, then click the square button at the right end of its its title bar.*
- *To temporarily re-open a collapsed pane, click where you see it's name pinned to the edge of the DMD window. Moving the cursor away from the pane, closes it automatically.*

On-Screen Presentation Creation
Top Toolbar Icons

 Save changes	Saves your work
 Delete Selected Object	***Deletes*** from your layout the one object or action that is selected on the workspace.
 Zoom In	***Increases*** the degree of magnification by 10 percentage points per click.
 Zoom Out	***Decreases*** the degree of magnification by 10 percentage points per click.
 Fit to View	***Increases or decreases*** the degree of magnification by whatever percentage amount is necessary to fit the entire screen object inside the visible workspace.
 Incremental Zoom	***Increases or decreases*** the degree of magnification by 1 percentage point per click, depending respectively on whether you click the arrow head that points up or the arrow head that points down. Alternatively, enter any percentage value in the unlabeled field, then press **Enter**.

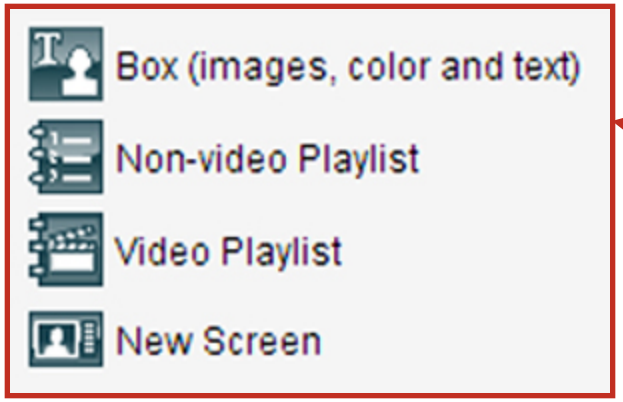
On-Screen Presentation Creation
Objects/Actions Panel

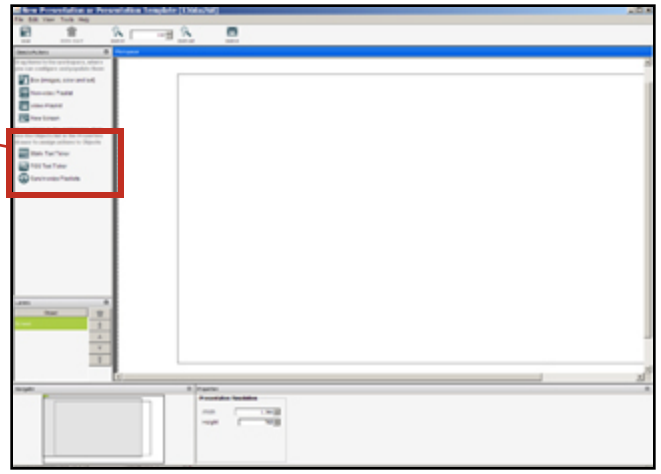
Box object icon

Non-Video Playlist object icon

Video Playlist object icon

Screen object icon

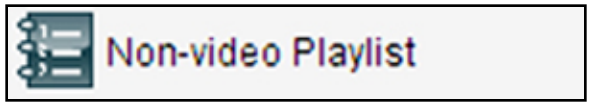
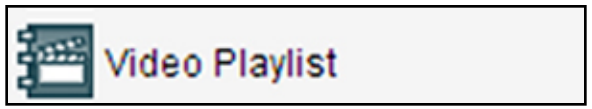




Creates the **representation of the DMP display** that yourscreen zones. ***Layouts cannot contain any more than one screen object at a time***. If you drag a second screen object to the workspace, Digital Media Designer asks you if you really want to overwrite the active screen in your layout. If you do overwrite the active screen, you simultaneously delete all of its configured screen zones and actions.

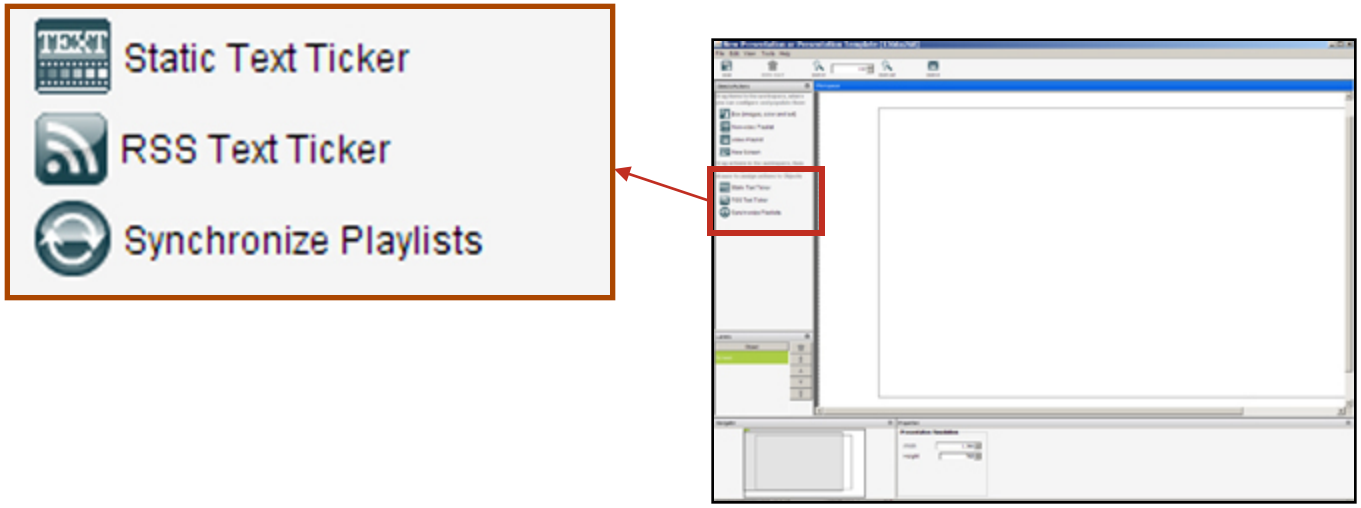
To start using a **Screen object** , **box object**, **Non-video Playlist**, ***and*** **Video Playlist objects *drag*** *an icon to the workspace.*

On-Screen Presentation Creation
Understanding Objects

	***Creates*** a **new screen zone** and a new layer where you can show one image file (**jpeg, gif, or png**), one **text** string, one background color, or a combination of them. (Other file types are not supported.) The text string can be static and presented without any visual effects when you show it on a DMP display or you can show it in the form of a ticker.
	***Creates*** a **new screen zone** and a new layer where you can arrange and play back in any combination an ordered sequence of **JPEG, png or gif** files, **SWF** files, and **pages on web servers**. Non-video playlists can be especially versatile when you use them as a background in your layouts.
	***Creates*** a **new screen zone** and a new layer where you can arrange and play back in any combination an ordered sequence of video files (**MPEG-1 and MPEG-2**), **SWF** files, **JPEG, png, or gif** files, **multicast video streams**, and **pages on web servers**. ***A layout cannot contain any more than one multimedia playlist object at a time.***

Note: To populate or configure each of these objects ensure that it's layer is selected then make selections in the Properties panel.

On-Screen Presentation Creation
Understanding Actions

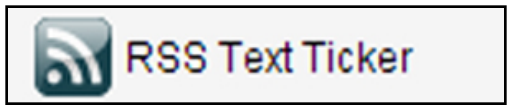


Text ticker behaviors that you can assign to any of the box objects in your layout. To show a ticker on a DMP display, you must ***associate*** a **ticker action with** a **box object** that is already part of your layout.

You can associate only one ticker at a time with any box object.

Notes: 1) To start using **Static Ticker**, **RSS Ticker**, ***or*** **Synchronize Playlists** ***drag*** *an icon to the workspace.*
2) To associate the ticker action with a box object, select one from the Box list in the Properties panel.

On-Screen Presentation Creation
Understanding Actions

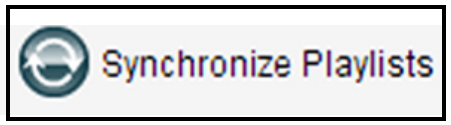


An **RSS ticker *displays*** text from an RSS feed that you specify in the Properties panel while the corresponding RSS ticker object is selected. Your DMP will check the RSS feed and update the ticker every 15 minutes.

Notes- The workflow to use a ticker action is as follows:

1. *To add the potential for a ticker to your layout, drag a ticker action to the workspace.*
2. *Do one of the following in the Properties panel while the ticker action is selected:*
   - *Static Ticker action—Enter text for your ticker in the Text field.*
   - *RSS Ticker action—Enter the RSS feed source URL for your ticker in the RSS URL field. The feed that you specify should be one that does not include any file enclosures.*
3. *To associate the ticker action with a box object, select one from the Box list in the Properties panel.*

On-Screen Presentation Creation
Understanding Actions



Synchronization a behavior that you can assign to two playlist objects (and therefore to two screen zones and two layers) in your layout. ***Binds*** the **selected playlists together** in the sense that you will synchronize the **playback** transitions between **media objects in the primary playlist** and the **playback** transitions between **media objects in the secondary playlist**—no matter how asynchronous those transitions would be ordinarily.

The playback timing from the primary playlist object will override the playback timing for the secondary playlist object.

Note:

1) *To add the potential for synchronized playlists to your layout, drag a synchronize playlist action to the workspace.*
2) *To identify the primary playlist, select it from the Primary Playlist list in the Properties panel.*
3) *To identify the secondary playlist, select it from the Secondary Playlist list in the Properties panel.*

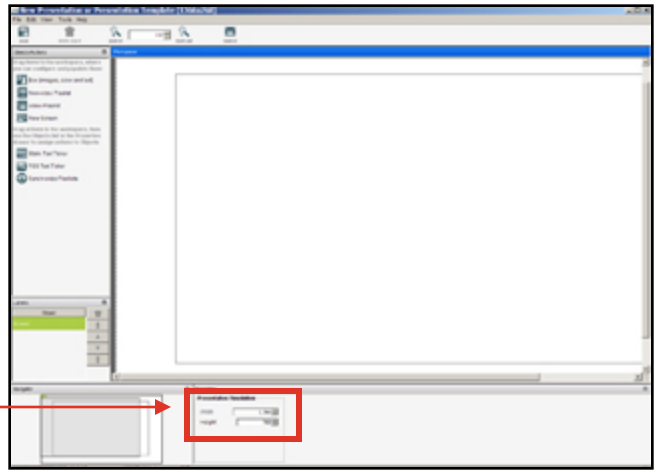
Presentation Creation in the Digital Media Designer

Step-by-Step Instructions

Return to Contents

On-Screen Presentation Creation
Creating New Presentations

- *Step 1* : In **Properties panel *use*** ***type-in boxes*** to enter your display resolution values in pixels.



Notes:

1) You can choose vertical or horizontal design from our templates, resave file as presentation, and than apply modifications to text and media files.

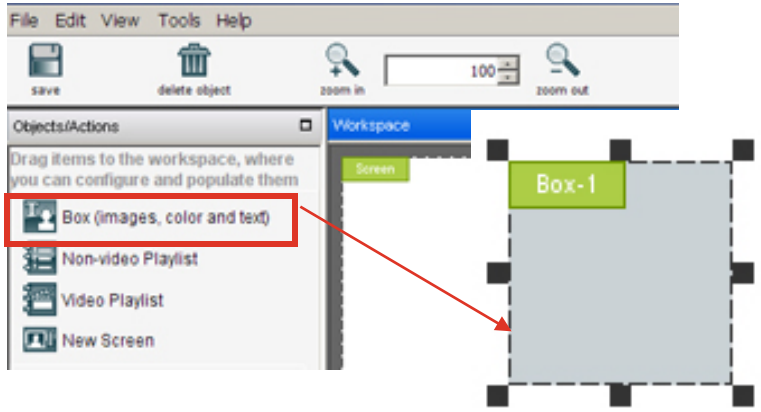
*2) It is **recommended** to **SAVE** every step*

On-Screen Presentation Creation
Adding a Background Image

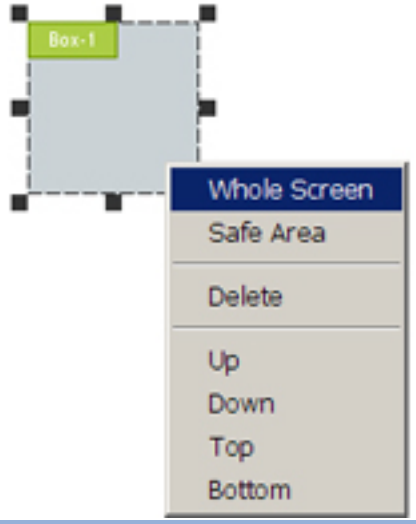
- *Step 2* : ***Drag*** the "**Box**" icon to the workspace. This box object will contain the background image that we will upload after.

  Hit the **SAVE** 

  button.



- *Step 3* : Specify location for the background image using arrows or insert values into type in boxes. If you want to apply background image to the whole screen, you can ***right-click*** with the mouse button and choose the "**whole screen**".



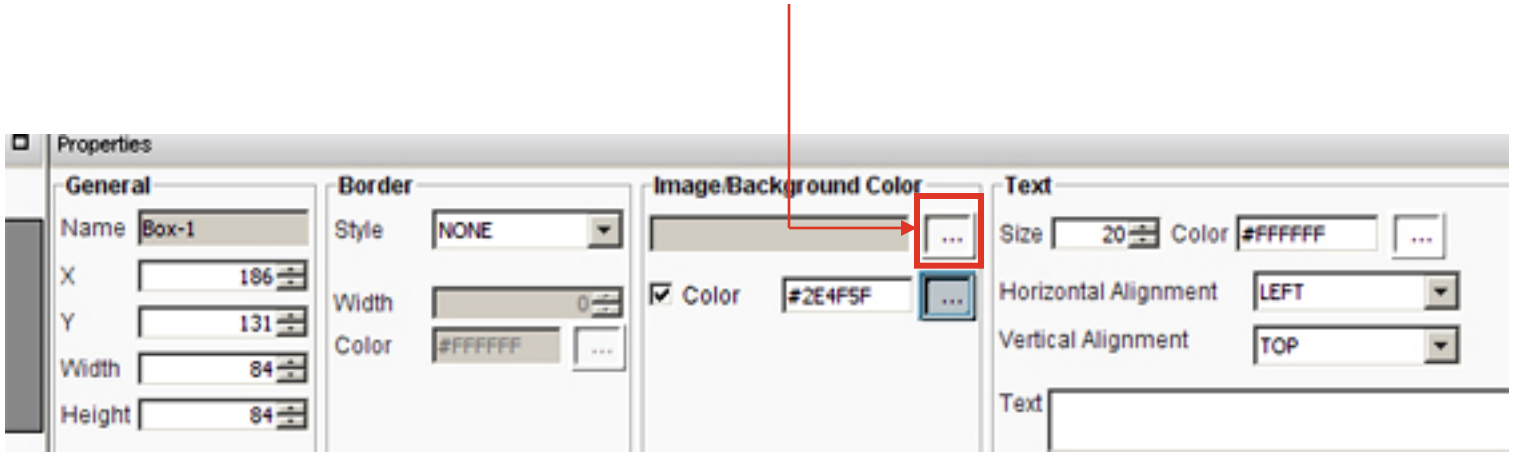
- **SAVE your work.**

Note: 1) Box object is a placeholder.

2) Box object holds one item at a time: it can be text or image.

On-Screen Presentation Creation
Adding a Background Image

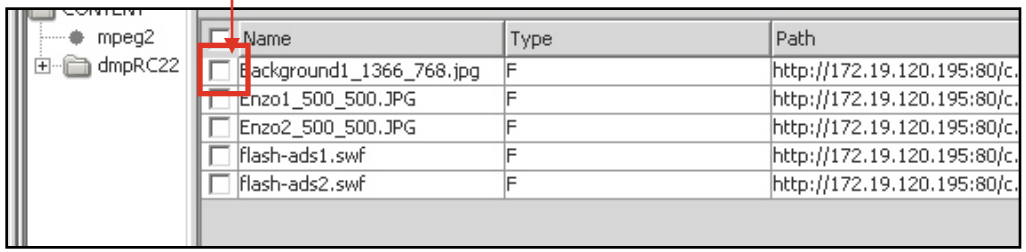
- ***Step 4*** : To upload image/media into box placeholder, with box/layer object selected ***click*** the **Browse Prompt** button in Properties Tab to open the Content Chooser dialog box.



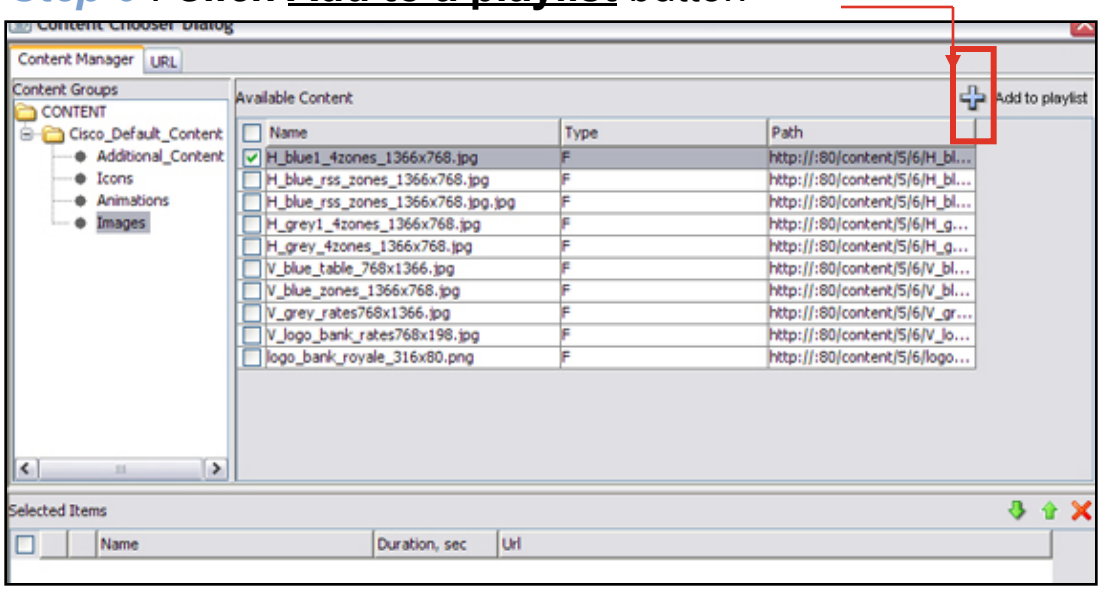
Note: You can also specify image background color by typing in your hex -decimal value or choosing with color picker. If you need transparent background like in transparent gif or png, uncheck the color check-box.

On-Screen Presentation Creation
Adding a Background Image

- *Step 5* : Choose an image you want to apply for background and ***check*** the check box next to file name.

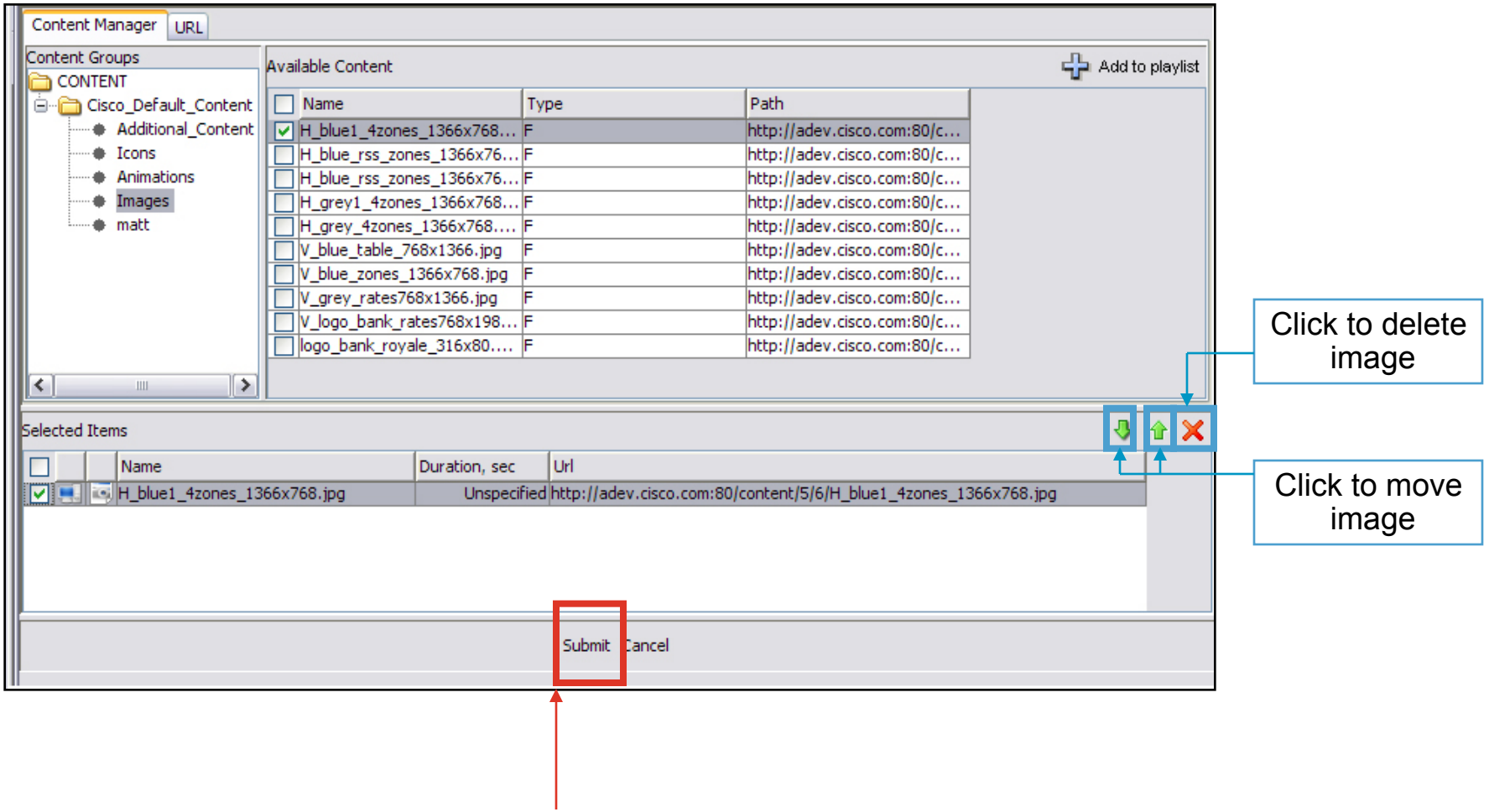


- *Step 6* : ***Click*** **Add to a playlist** button



Note: You can upload just one image at a time.

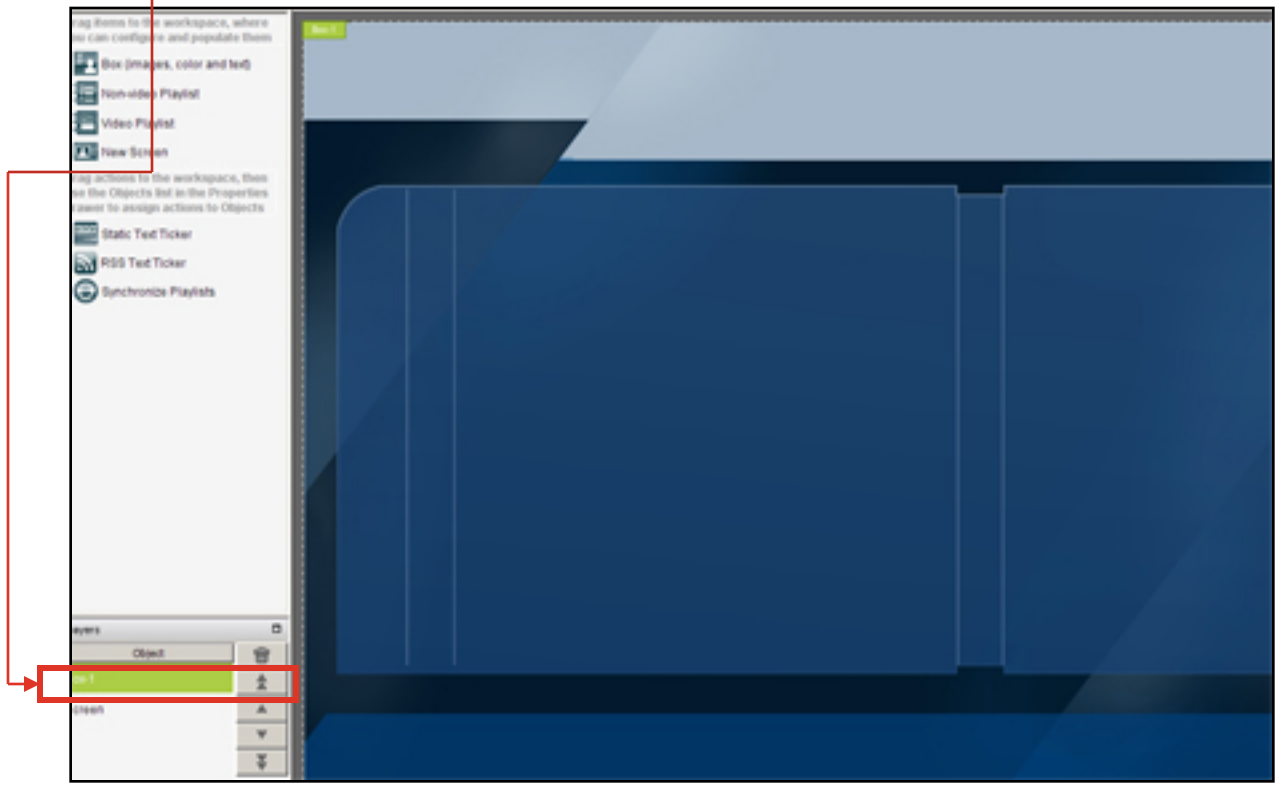
On-Screen Presentation Creation
Adding a Background Image



- *Step 7*: ***Click*** *the* **Submit** button to place the file in it's placeholder - box

On-Screen Presentation Creation
Adding a Background Image

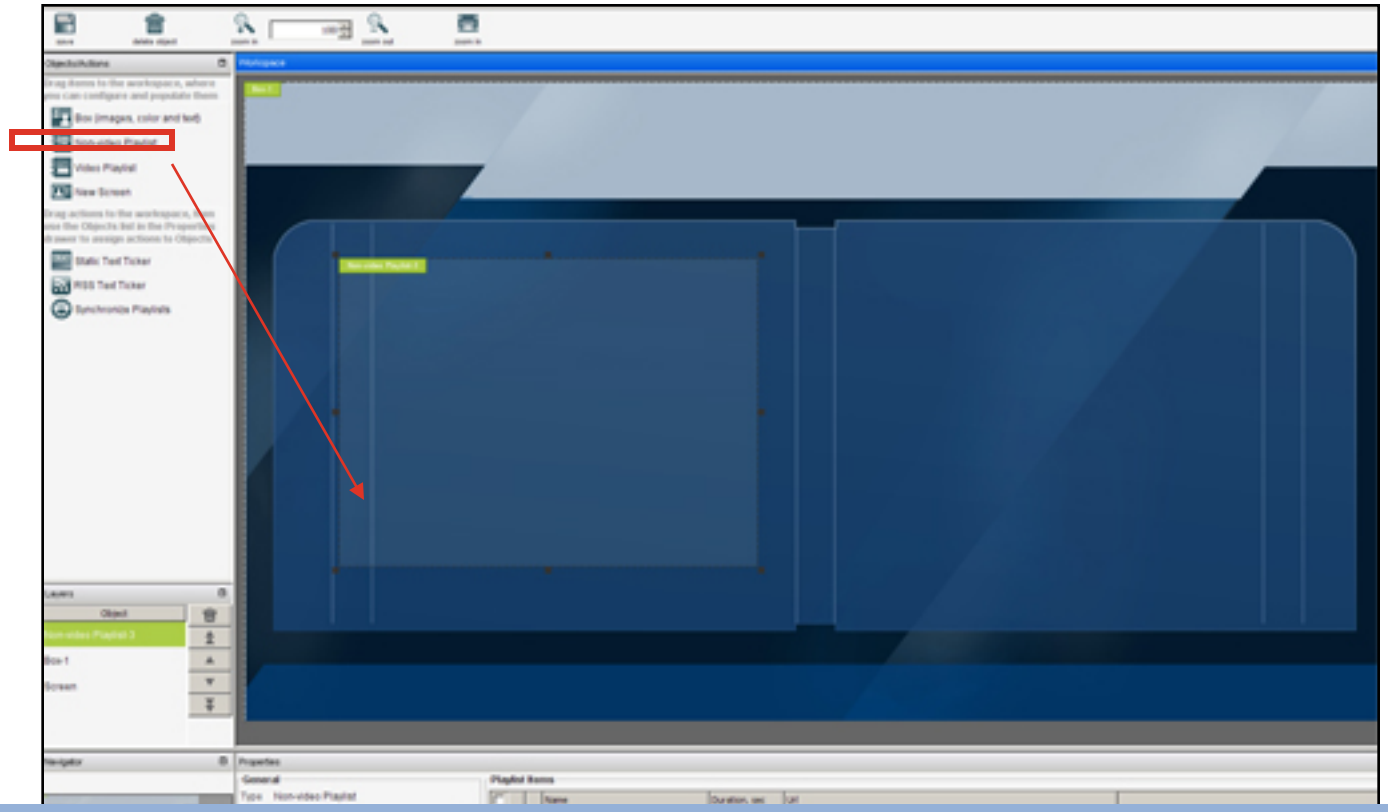
- After you hit the **Submit** button you'll see image preview in work area. When box placeholder is selected, it's layer is selected as well.
- **SAVE your work.**



Notes: 1)Only image files are rendered in work area. There is no preview for Flash movies, MPGs, etc.
2)If you place a JPEG image in a box object, be sure that the box object uses the same height and width in pixels that the JPEG file uses. If the box object is too small, the displayed JPEG image will be cropped.

On-Screen Presentation Creation
Adding Non-Video Playlist

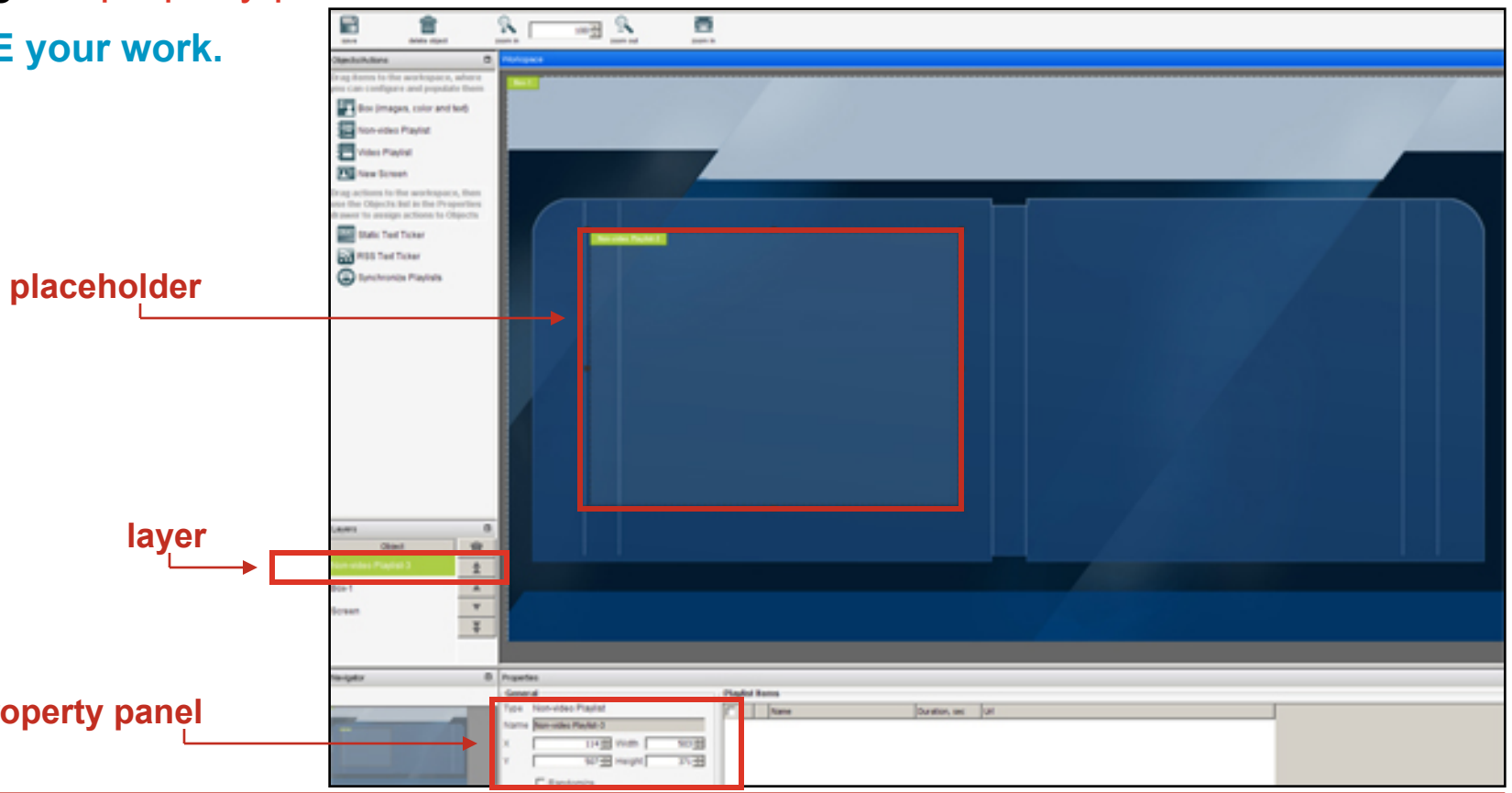
- ***Step 9***: ***Drag*** the **Non-video Playlist** to your flash movie zone.



*Notes: **Non-video Playlist** is a screen zone and layer where you can arrange and play back in any combination an ordered sequence of JPEG files, SWF files, and pages on web servers.*

On-Screen Presentation Creation
Adding Non-Video Playlist

- ***Step 10***: ***Adjust size and position*** of your **Non-video Playlist** placeholder using the property panel.
- **SAVE your work.**

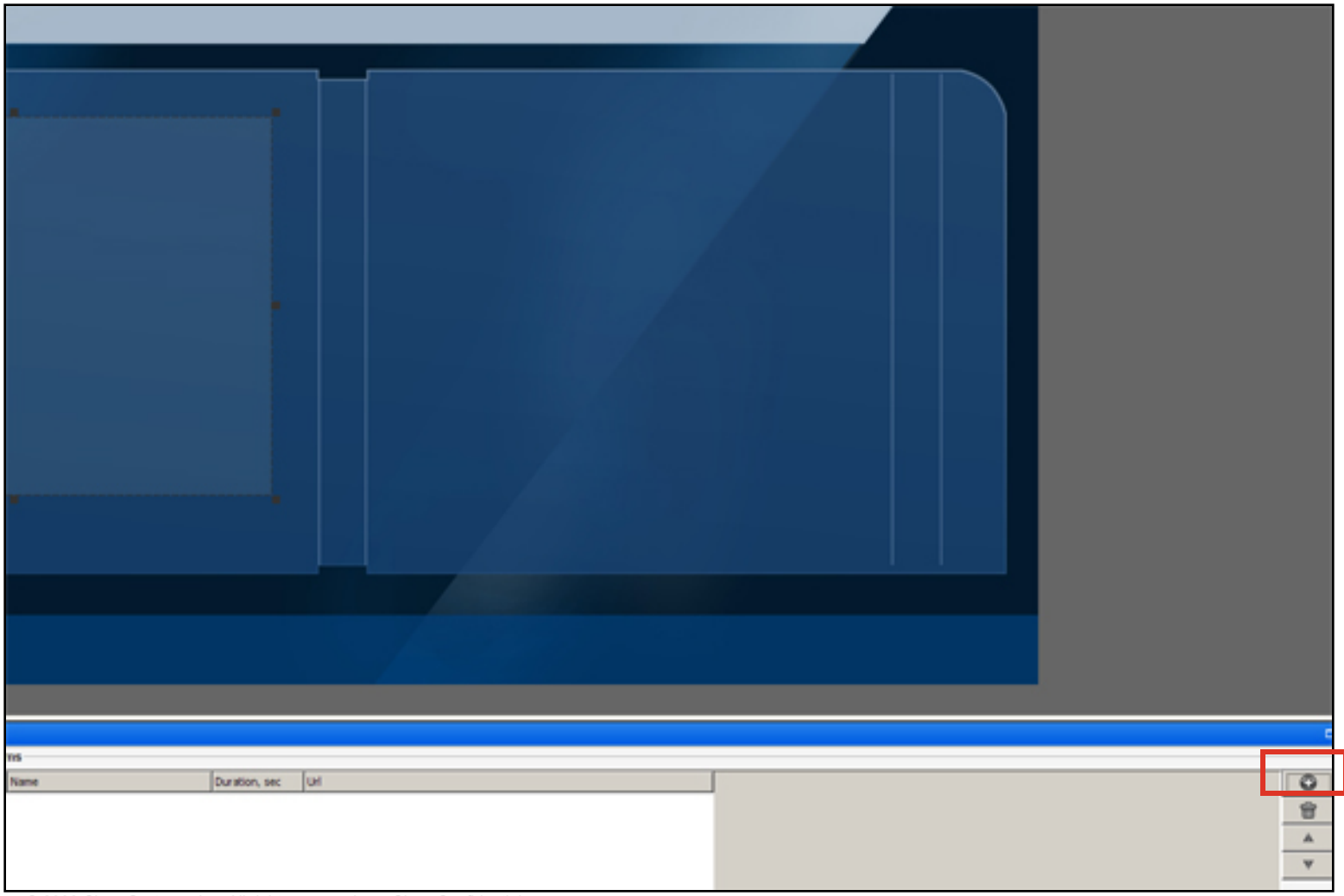


*Notes: When any item is selected, the layer that this item associated with is selected as well. If you need to **make changes or modifications** to any **media** file you can **select** it's **placeholder**, or **layer**.*

On-Screen Presentation Creation

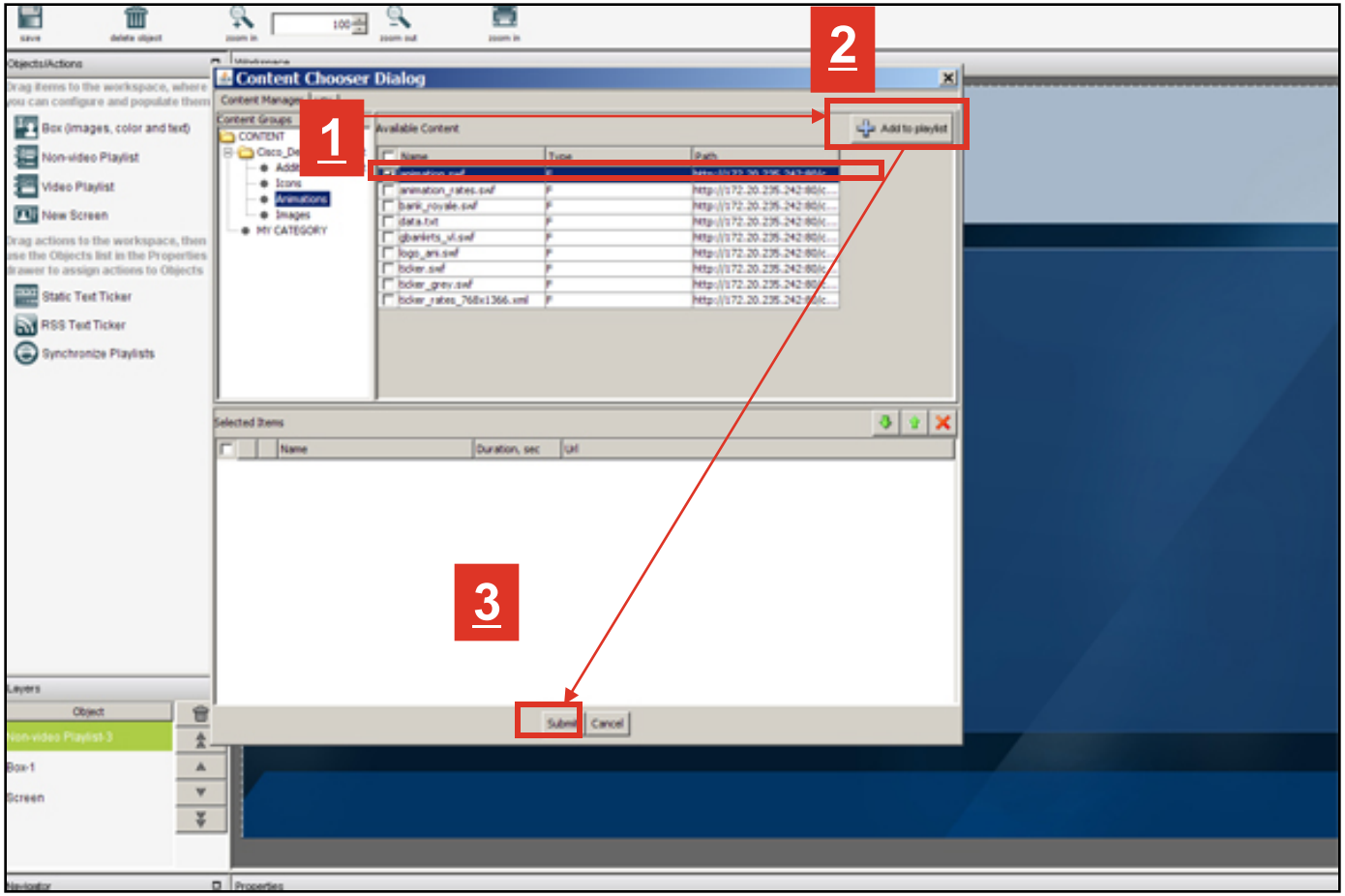
Adding Non-Video Playlist—Assigning Media Assets

- *Step 11*: ***Click*** on the "**plus sign**" icon to apply flash movie to it's **Non-video Playlist** placeholder.



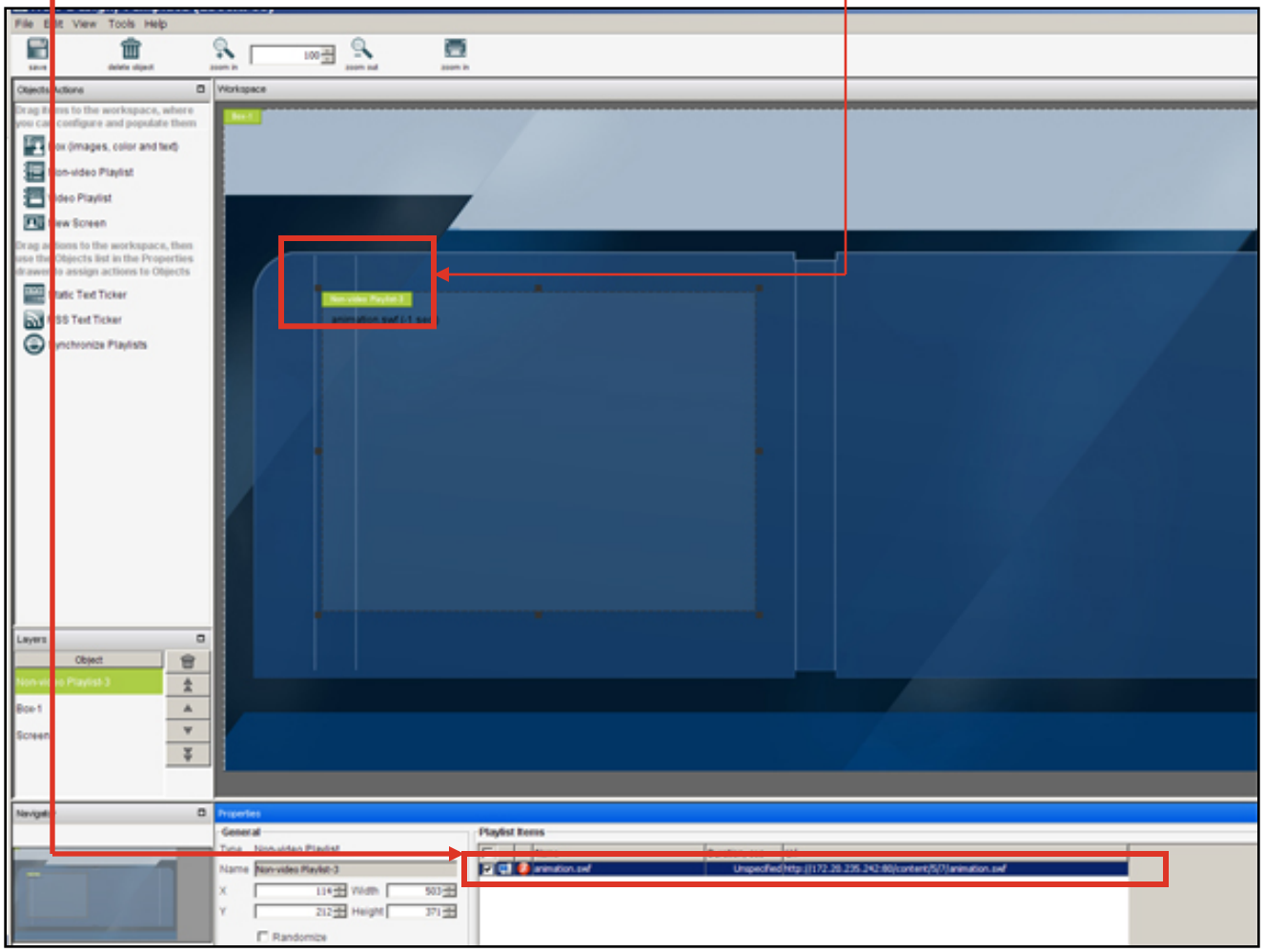
On-Screen Presentation Creation
Adding Non-Video Playlist—Assigning Media Assets

- ***Step 12***: ***Choose*** flash movie from animations folder **to upload** and follow steps:



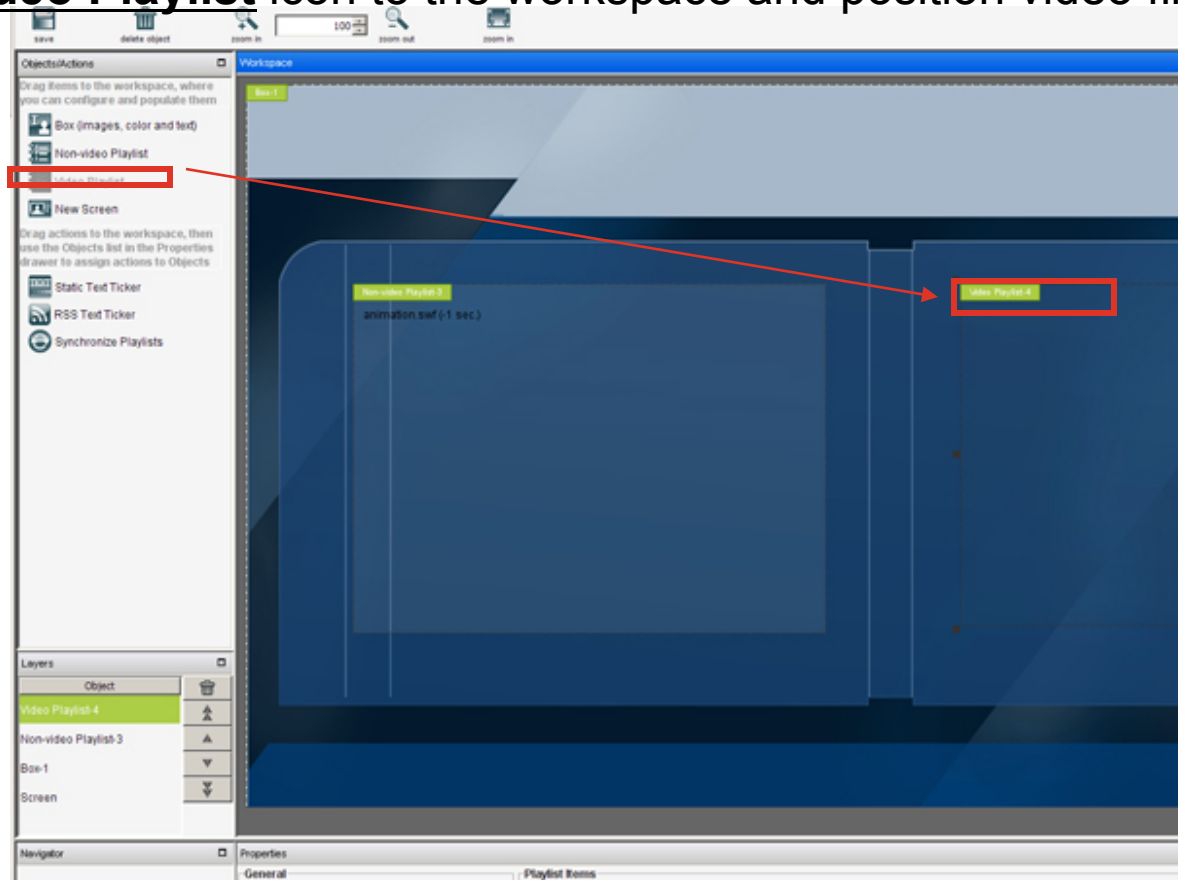
On-Screen Presentation Creation
Adding Non-Video Playlist—Assigning Media Assets

- ***Step 13***: Flash movie name and information shows up in **Non-video Playlist** placeholder and in **Properties Panel.**
- **SAVE your work.**



On-Screen Presentation Creation
Adding Video Playlist

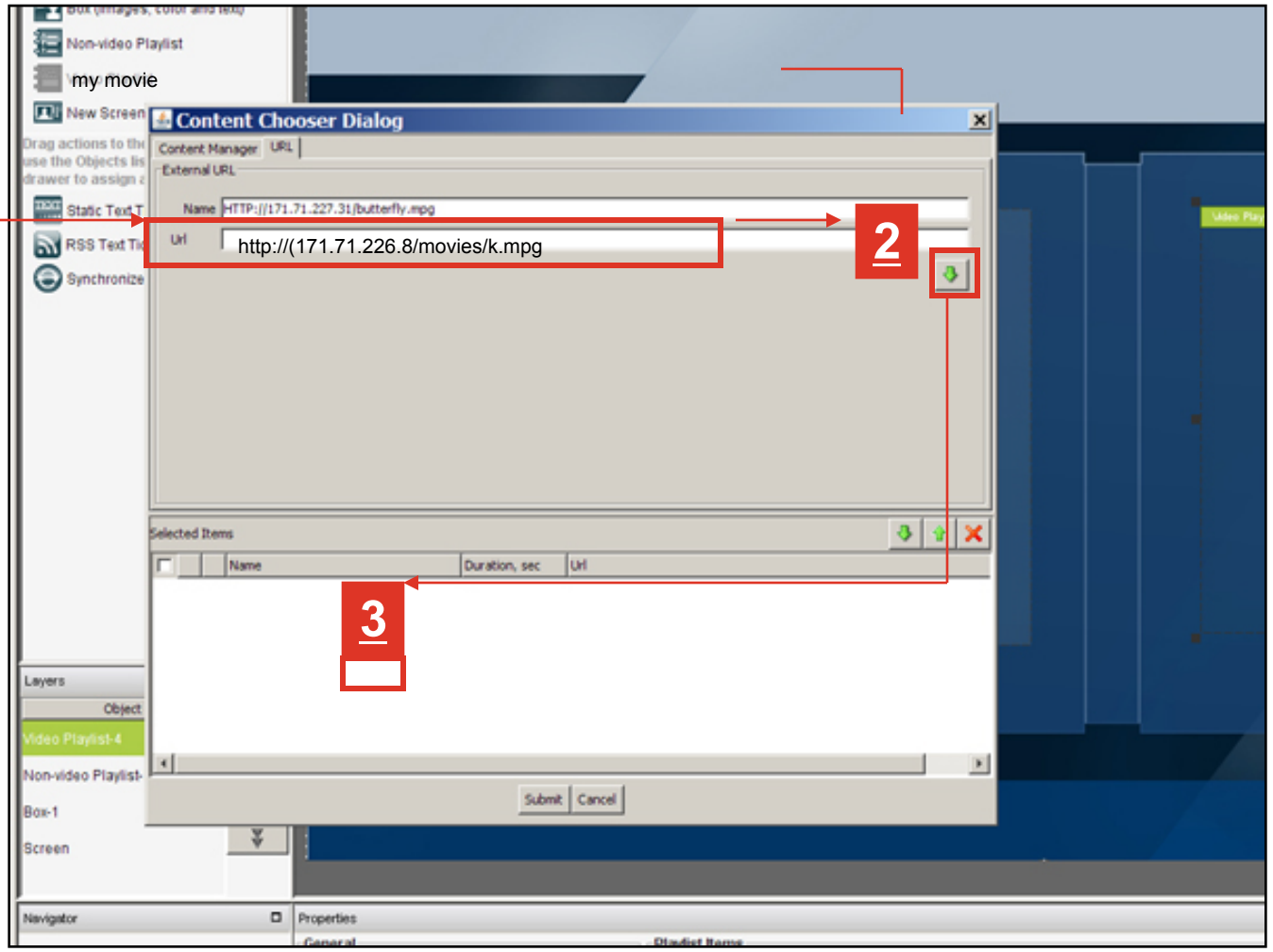
- ***Step 14***: Drag the **Video Playlist** icon to the workspace and position video file.
- **SAVE your work.**



Note:
Video Playlist *is a screen zone and layer that shows all Non-video Playlist qualities plus MPEG video and multicast video streams playback*

On-Screen Presentation Creation
Adding Video Playlist—Applying Media Assets

- ***Step 15***: To ***upload*** video file you can use **Content Manager** panel (same way as you used for flash movie) or **URL** tab.



To upload video file:

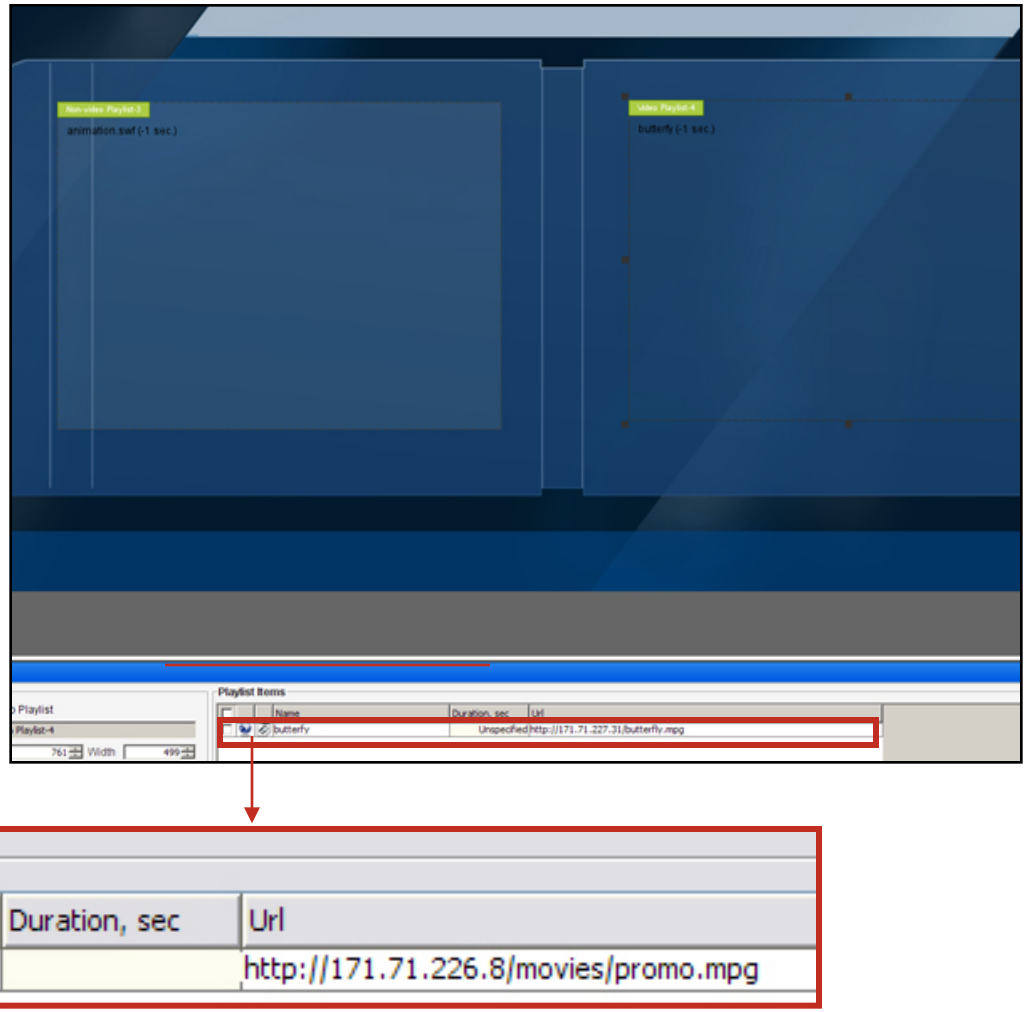
1) Type in movie name and URL

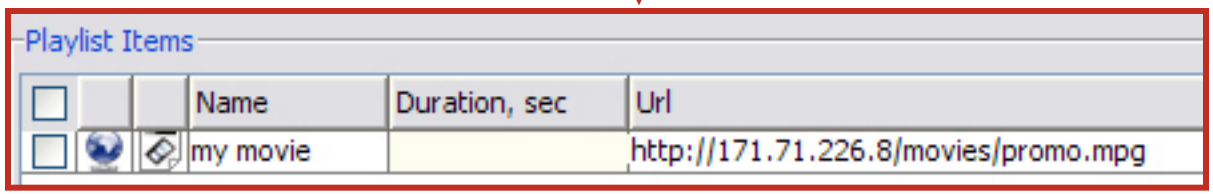
2) Click on the Down Arrow

3) Click the Submit button

On-Screen Presentation Creation
Adding Video Playlist—Applying Media Assets

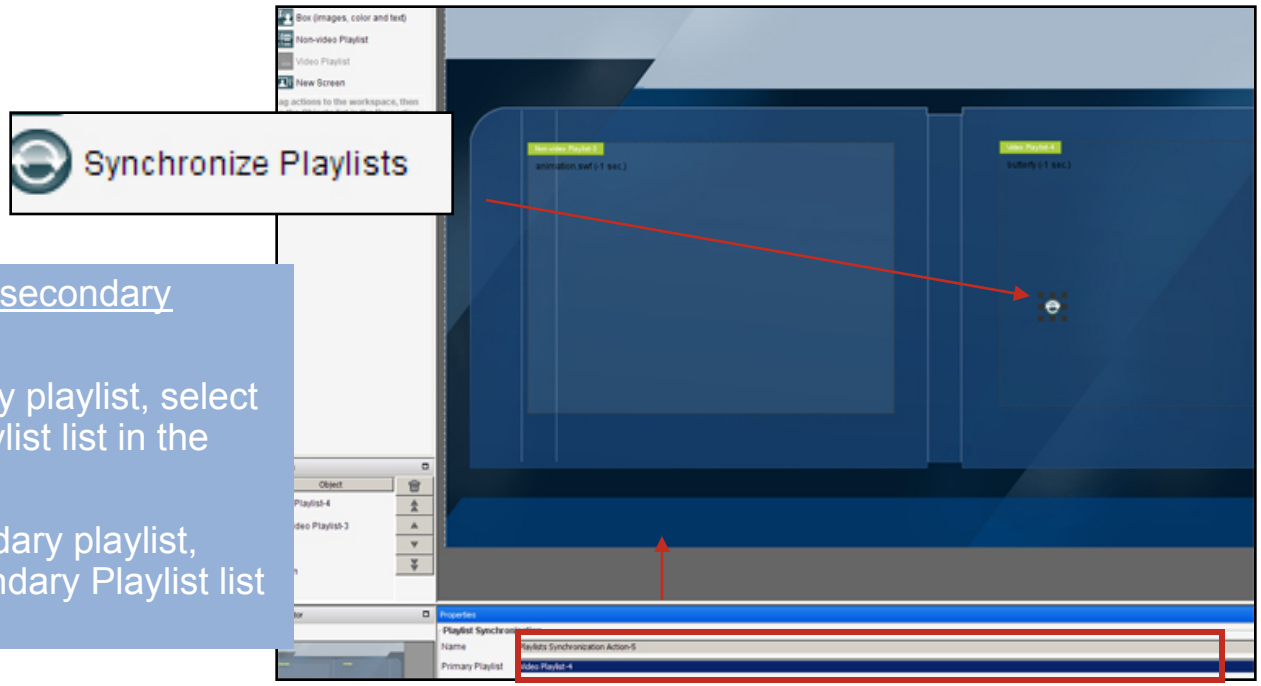
- Video **file name and information** shows up in **Video Playlist** placeholder and in **Properties Panel.**
- **SAVE your work.**





On-Screen Presentation Creation
Synchronizing Playlists

- ***Step 16***: To ***synchronize*** 2 playlists, ***drag*** a **synchronize playlists action** to the workspace.



Choose primary and secondary playlists here

To choose Primary and secondary Playlists:

1) To identify the primary playlist, select it from the Primary Playlist list in the Properties Tab.

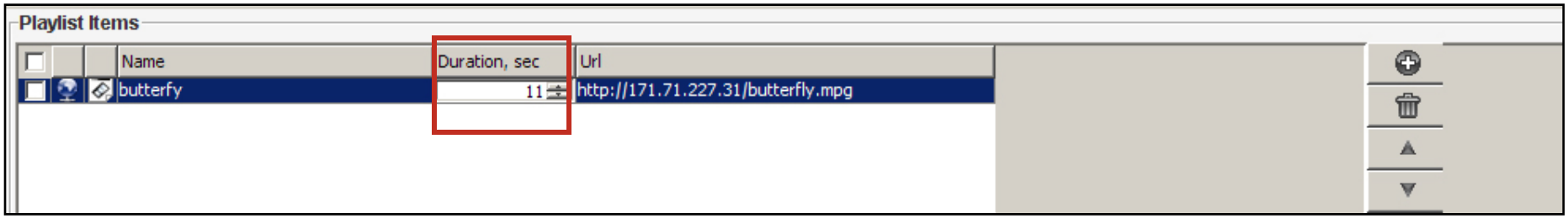
2) To identify the secondary playlist, select it from the Secondary Playlist list in the Properties Tab.

- **Pink arrow** appears on synchronization action icon indicating that the ***action has been applied***

- **SAVE your Presentation.**

On-Screen Presentation Creation
Synchronizing Playlists

- ***Step 17***: In Properties Panel ***define*** the **playback duration** for each entry that it contains.



Notes:
1)To populate and configure a playlist object, click it on the workspace,or select it's layer then make selections in the Properties panel. For example, the Properties Panel is where you define the playback sequence for a playlist and define the playback duration for each entry that it contains.

2)Ensure that your **playlists** *layers are the* **topmost**, *and* **background** *layer is on the* **bottom** *layer.*

On-Screen Presentation Creation
Adding Tickers

- ***Step 18*** : Add Ticker to your presentation

To add Ticker:

1) Drag box icon to work area.

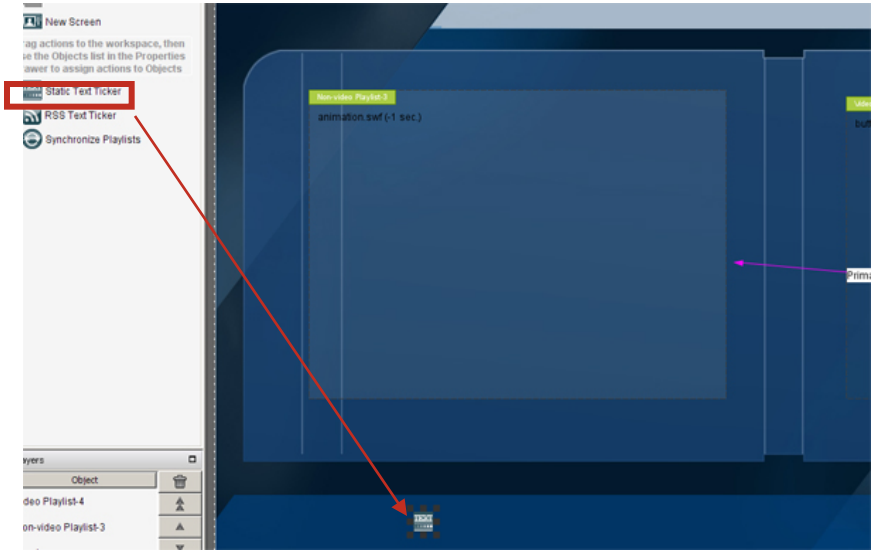
2) Adjust dimensions and positioning in the Properties Panel.

3) Choose/adjust the background color for Ticker using Properties Panel.

4) While box is selected drag the Static Ticker icon to apply Static Ticker action to box.

5) With Static Ticker Action selected, choose the box from Assign to Object drop -down list to associate this action with the box object.

6) Enter text for your ticker in the Text field.



4) Static Ticker action have been applied, but it's not associated with box yet

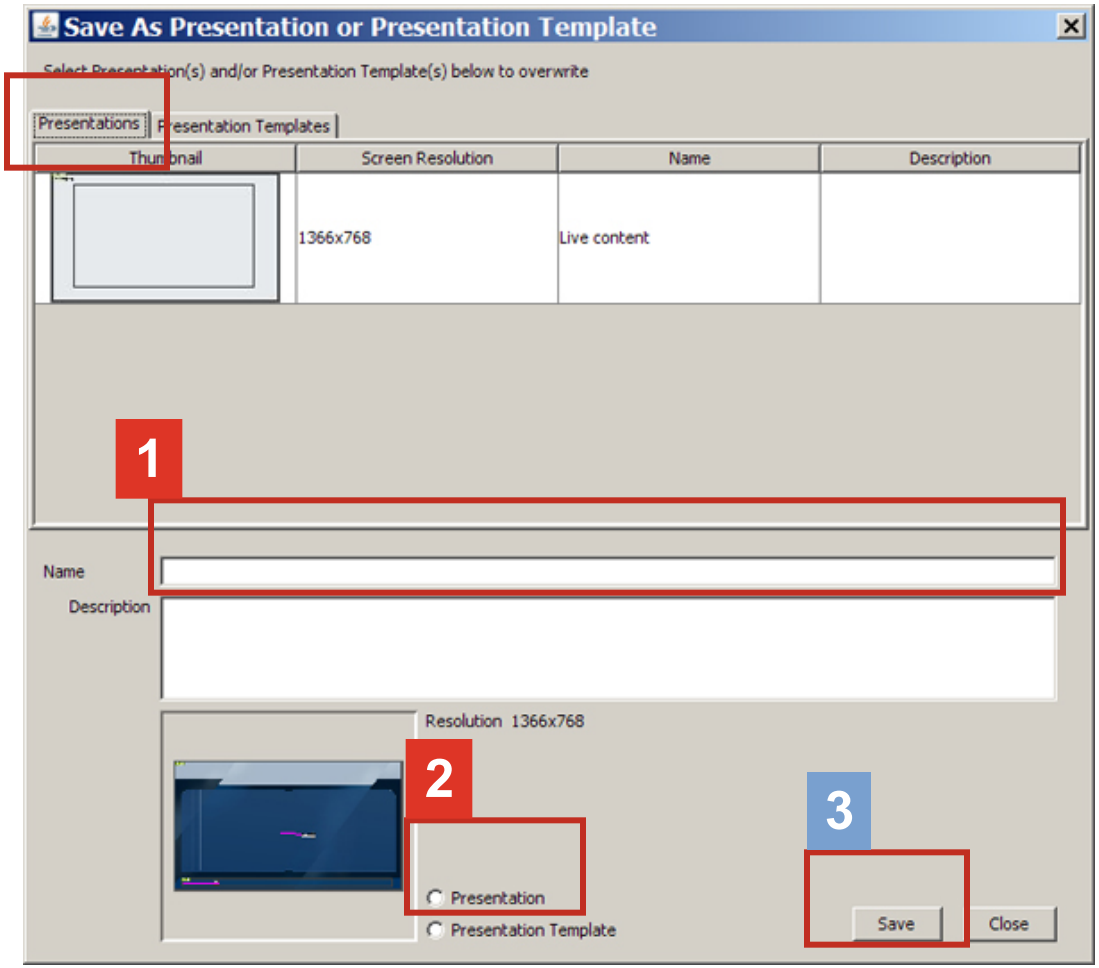
5) Static Ticker action has been assigned to the box. Pink arrow indicates that it have been applied to box 4.

- **SAVE your Presentation**

Note: You can associate only one ticker at a time with any box object.

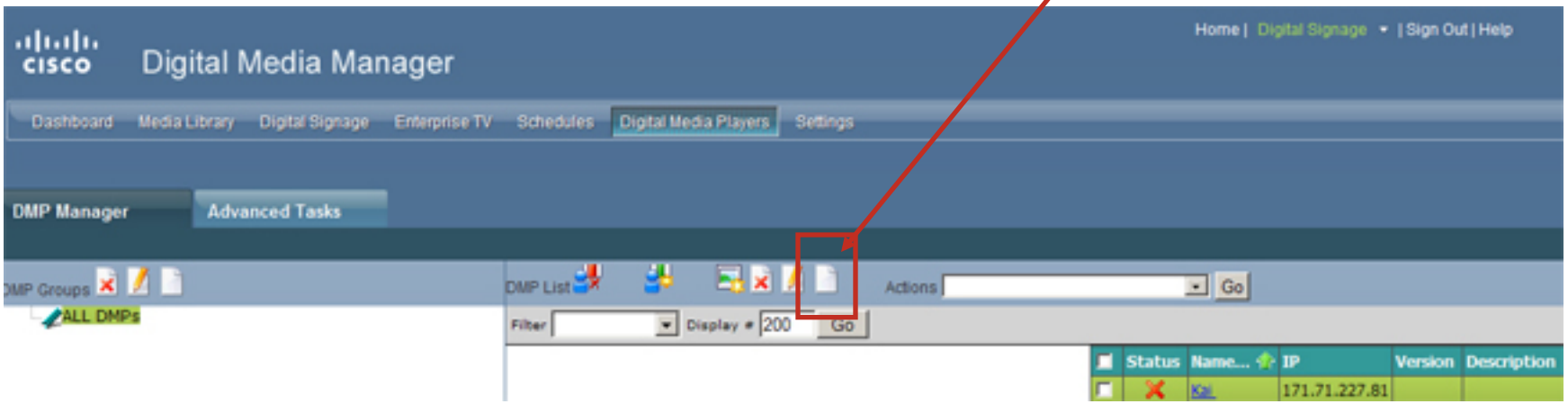
On-Screen Presentation Creation
Creating New Presentations

- *Step 3* : Save your file as Presentation in **Presentations** Tab: 1) enter presentation name, 2) select the **Presentation** radio button, 3) hit the **SAVE** button



On-Screen Presentation Creation
Publishing Presentations

- *Step 19* : To ***publish*** your presentation you need to ***register your DMPs***:
  1) open in **DIGITAL MEDIA PLAYERS** panel in DMM
  2) in DMP Manager click on Add New DMP button
  3) Enter IP and Mac address of your DMP
  4) Select your DMP from the list
  5) From Actions drop-down menu choose your presentation
  6) Click on **Go** button

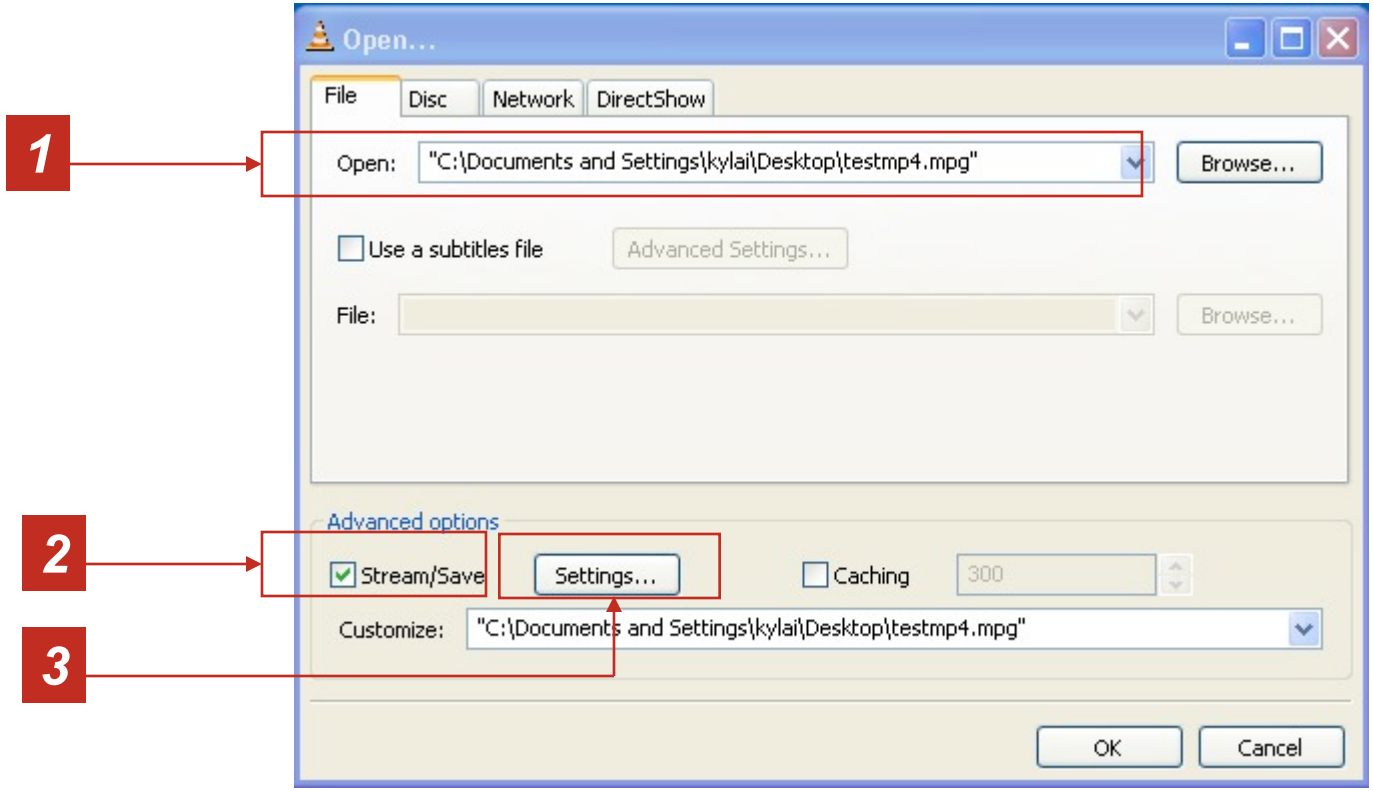


MPEG Creation

Return to Contents

VLC: Encoding to MPEG2-TS

- ***Step 1:*** Open the file you need to encode into MPEG-2
- ***Step 2:*** Check “Stream/Save” check-box
- ***Step 3:*** Click “Settings”

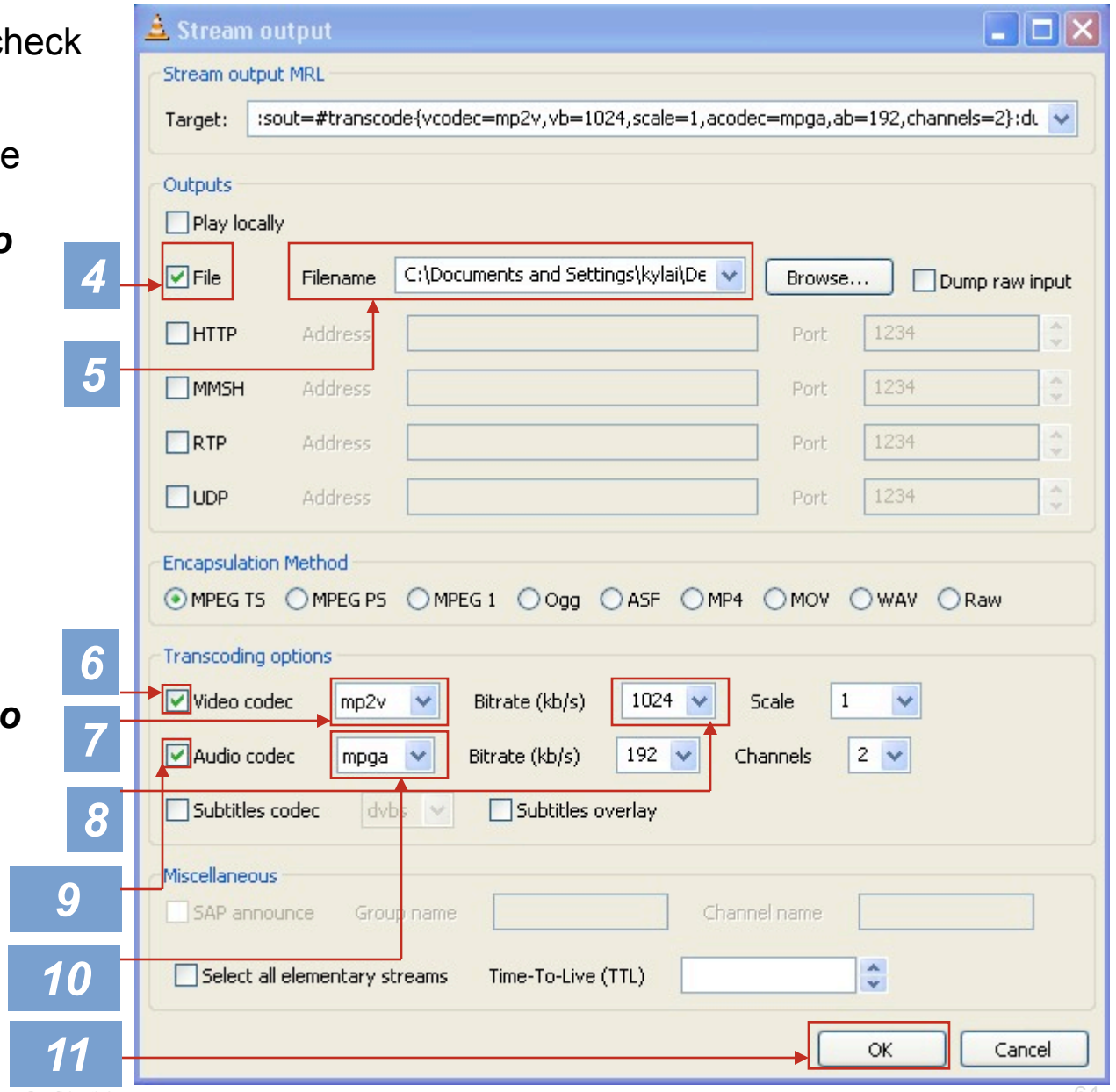


VLC: Encoding to MPEG2-TS

- ***Step 4:*** Check the ***file*** check box
- ***Step 5:*** Specify file name
- ***Step 6:*** Check the ***video codec*** check box
- ***Step 7:*** Choose ***mp2v*** from the drop-down menu
- ***Step 8:***

Choose the ***bitrate***:
- for HD: 12-15 Mbit/sec
- for SD: 5-8 Mbit/sec

- ***Step 9:*** Check the ***audio codec*** check box
- ***Step 10*** Choose ***mpga*** from the drop -down menu
- ***Step 11:*** Hit ***OK***

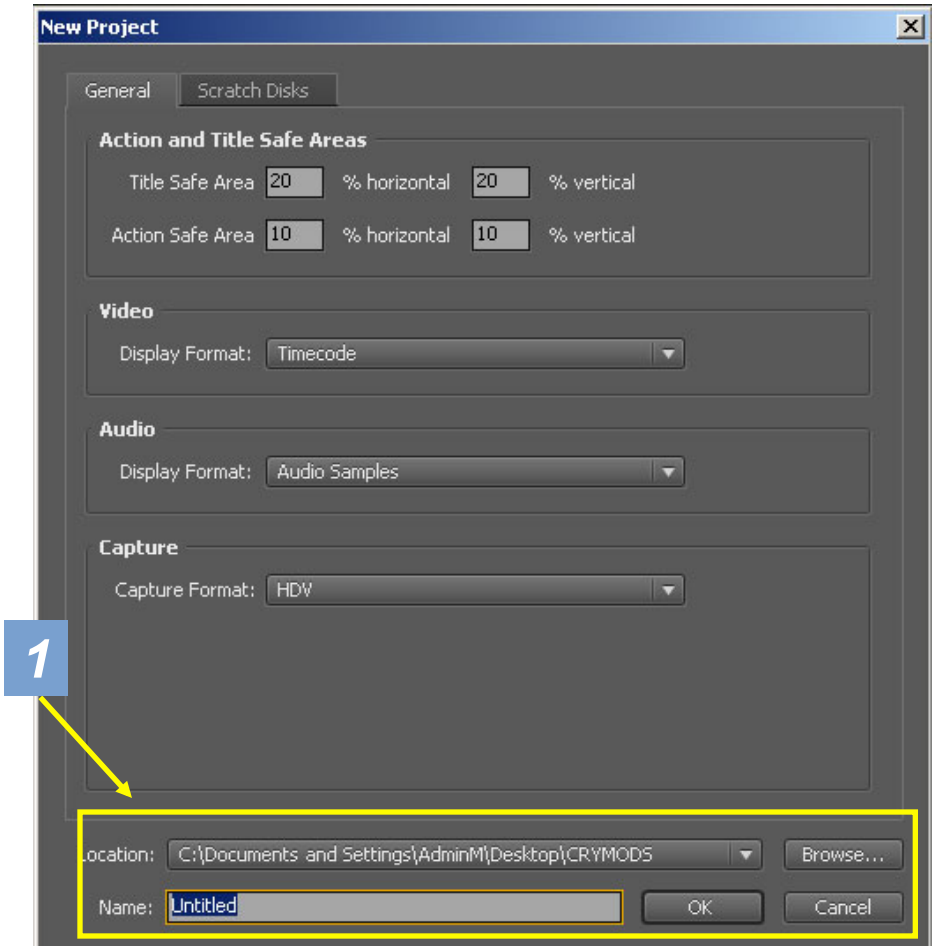


Adobe Premiere Pro CS4: MPEG-2 Creation

- In Adobe Premiere, it's best to use existing presets for this project:
  - When you open a new project, please choose Adobe HDV 720p 30 or HDV 1080p 30.
  - And then, in Export Adobe Media Encoder please choose MPEG2 for format, Range: render entire sequence, Preset:Custom; Video Codec :MainConcept MPEG Video;
- Quality: Best; TV standard: NTSC;
- Frame Width for this example: 1280p;
- Frame Height: 720px;
- Frame rate: 29.97 drop frame;
- Field Order: None (progressive)
- Pixel aspect ratio: Square Pixels
- Profile: Main
- Level: High level
- Bitrate Encoding : CBR, or VBR (for VBR numbers should not exceed Max bitrates specified below)
- Bitrate for HD: 10 Mbit/sec - 15 Mbit/sec (Max)
- Bitrate for SD: 3 Mbit/sec - 5 Mbit/sec (Max)
- Multiplexing: TS (transport stream)

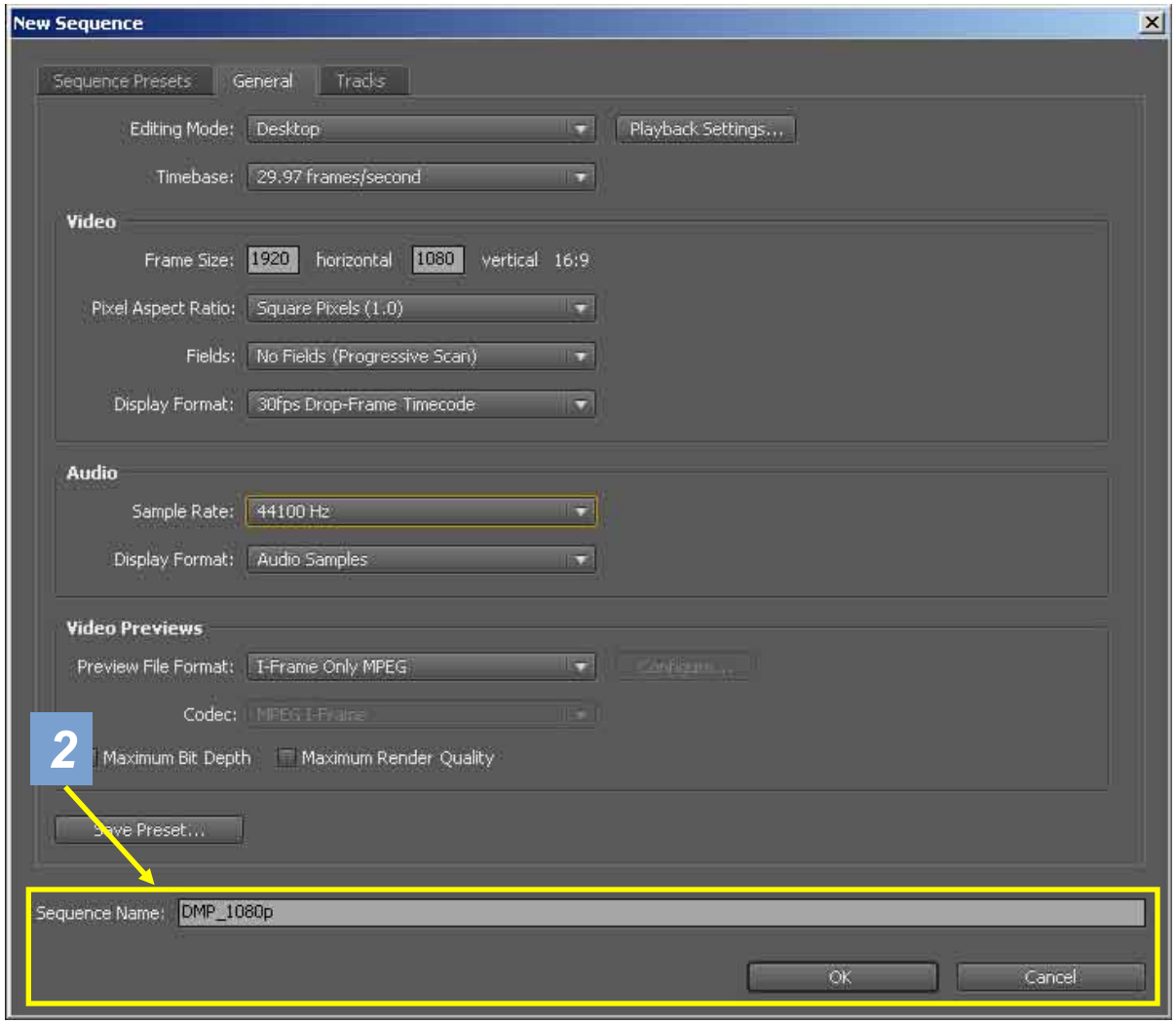
Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 1: Within*** "General" tab Name your project.



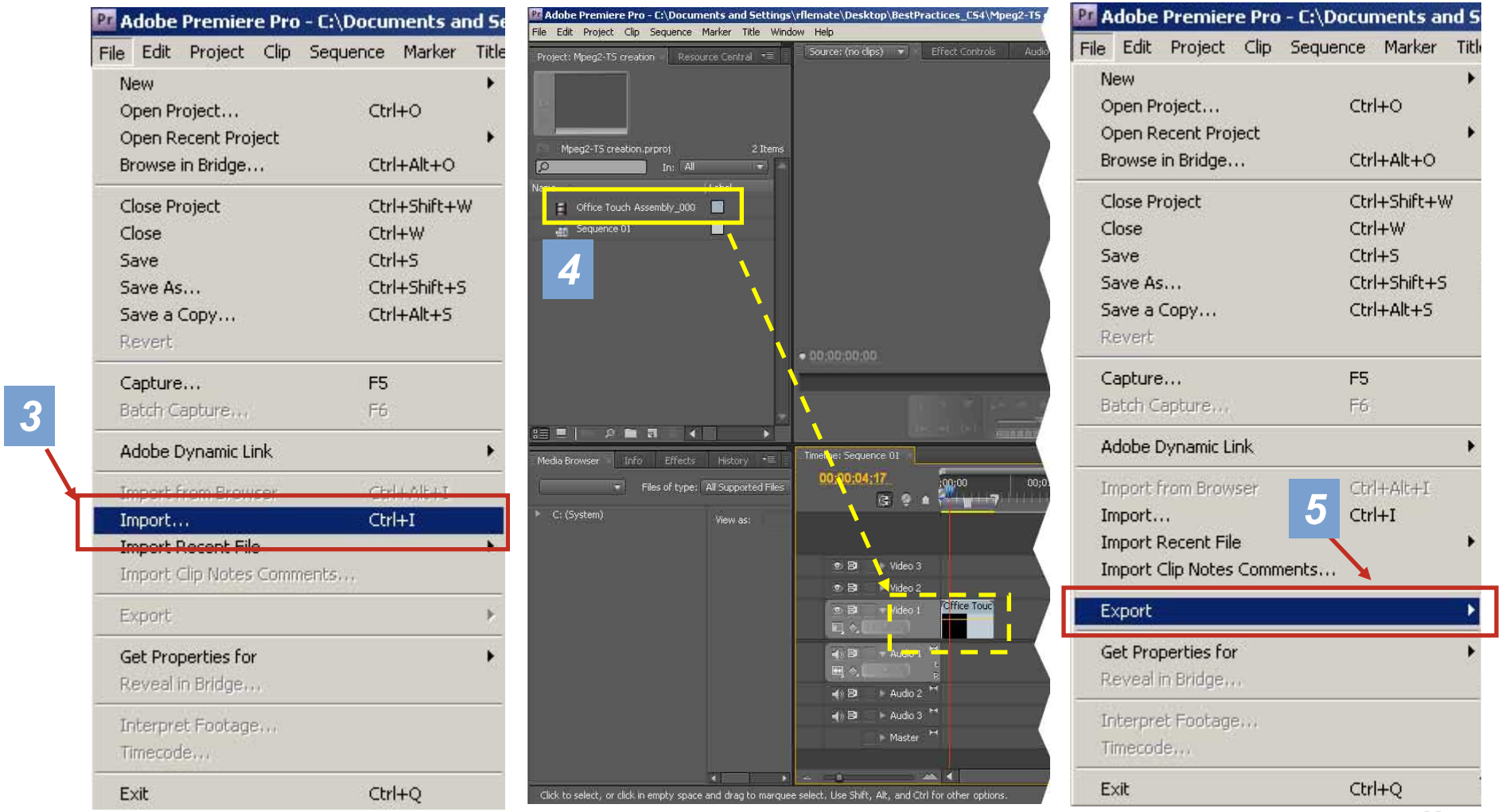
Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 2: Within*** "General" tab Name your Sequence this is NOT naming the file.



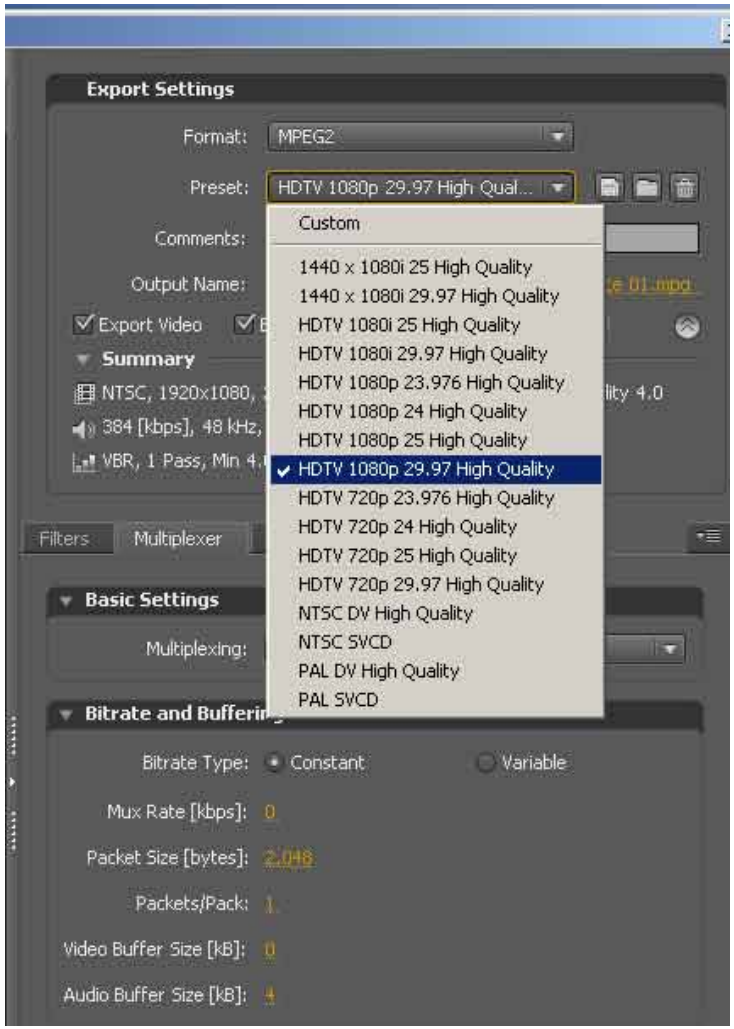
Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 3:*** ***import*** your footage
- ***Step 4:*** Drag imported footage into the timeline for “Video 1”.
- ***Step 5:*** Select File>Export→ this will open the Adobe Video Encoder.



Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 6*: Select** *File>Export. In the dialog box choose a preset.*

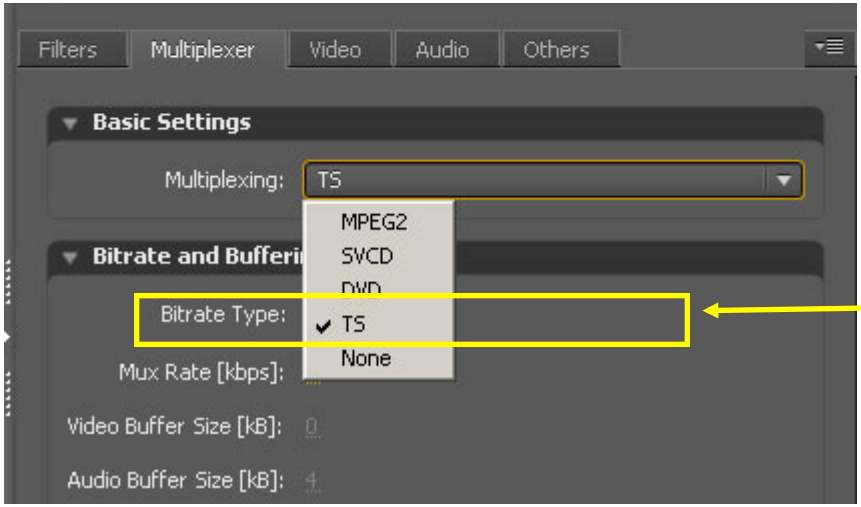


Don't worry about the preset, you will alter the parameters in

Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 7*: *The Multiplex, Video, and Audio tabs:***

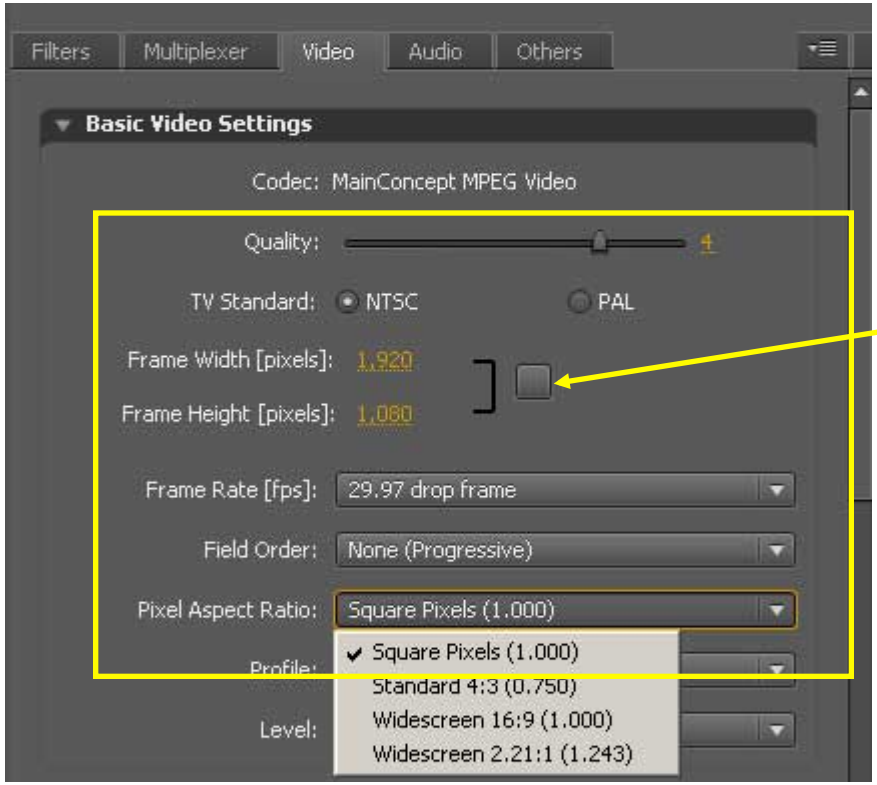
Multiplex



Select the "TS" (transport stream) setting. In CS4, bitrates are now handled in the Video tab.

Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 8a*: *The video tab.***



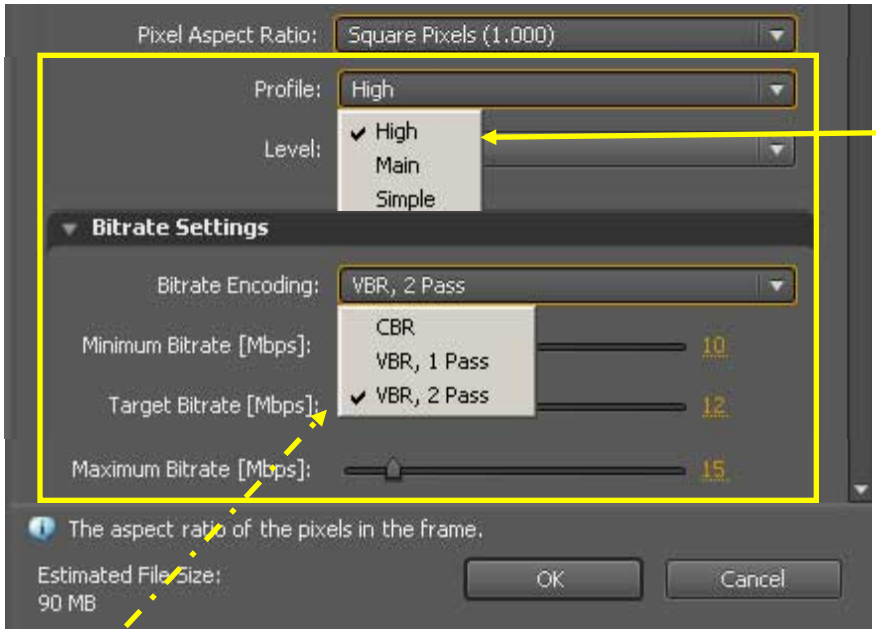
Video 8a

These are the general video settings.

Be sure to uncheck the constraint and match your Premiere project's custom width and height.

Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Steps 8b & 8c*: *The video tab.***



Video 8b, 8c

Although not a bitrate par setting, selecting the proper profile impacts the range of bitrates available. For 720p video output and greater, choose the "High" profile for DMP playback.

VBR, 1 Pass is a good compressor, "2 Pass" will create a smaller file size and doubles rendering time.

A good bitrate range is between 8MB/s to 18 MB/s.

As with any data limiter, test your results for visual quality.

Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 9*: Audio tab: Select MPEG format**

Audio tab:

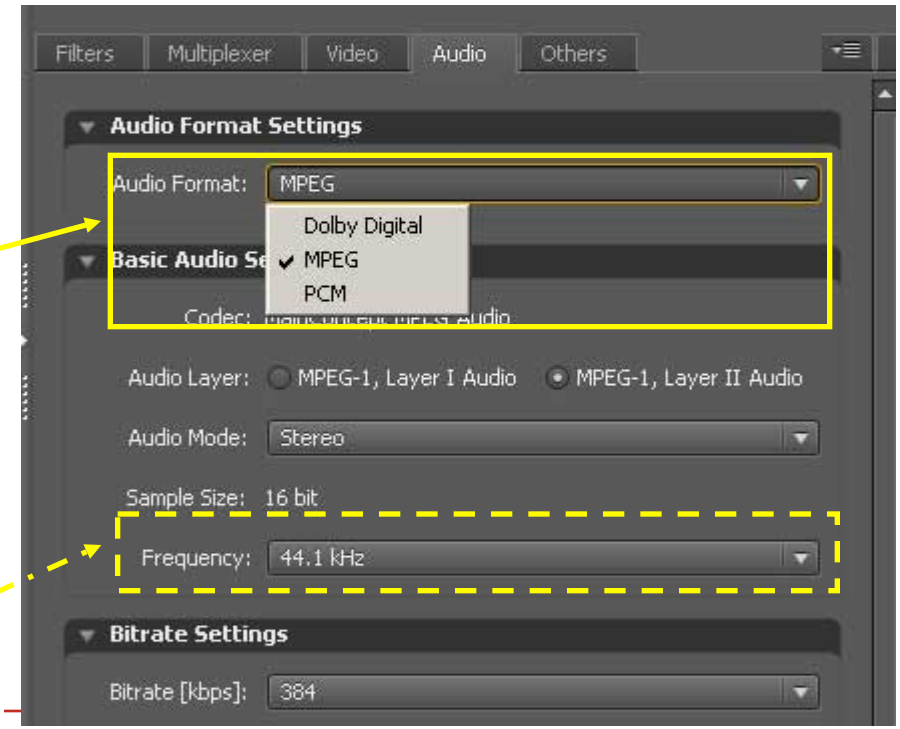
Whether audio is used or not, include audio output to ensure proper functionality to the transport stream (TS).

Audio Frequency Guide:

48.0 kHz: DVD quality.

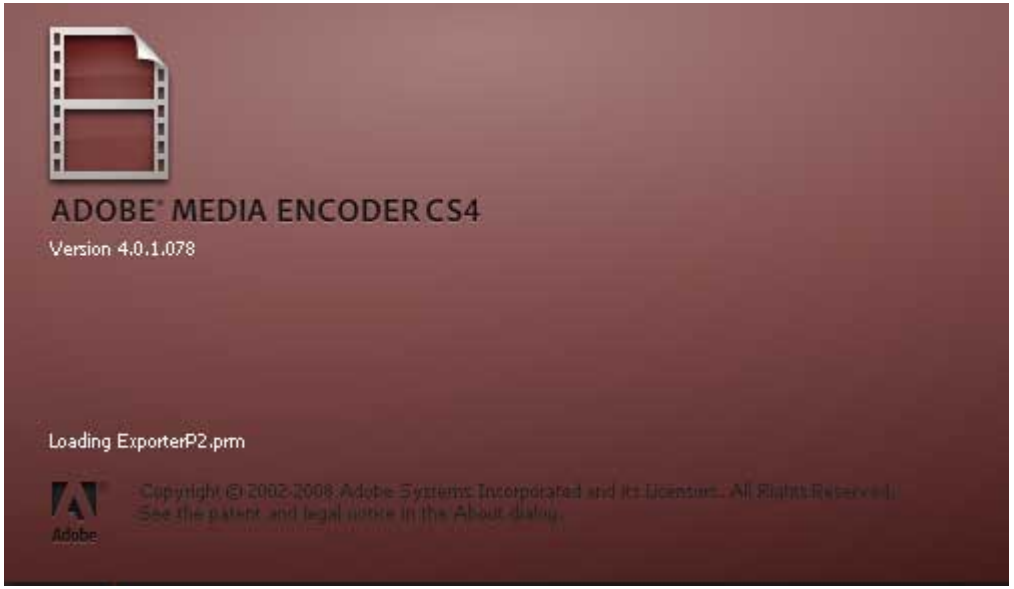
44.1 kHz: CD quality.

32.0 kHz: Generic



Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 10*: Clicking "OK" will launch Adobe Media Encoder.**

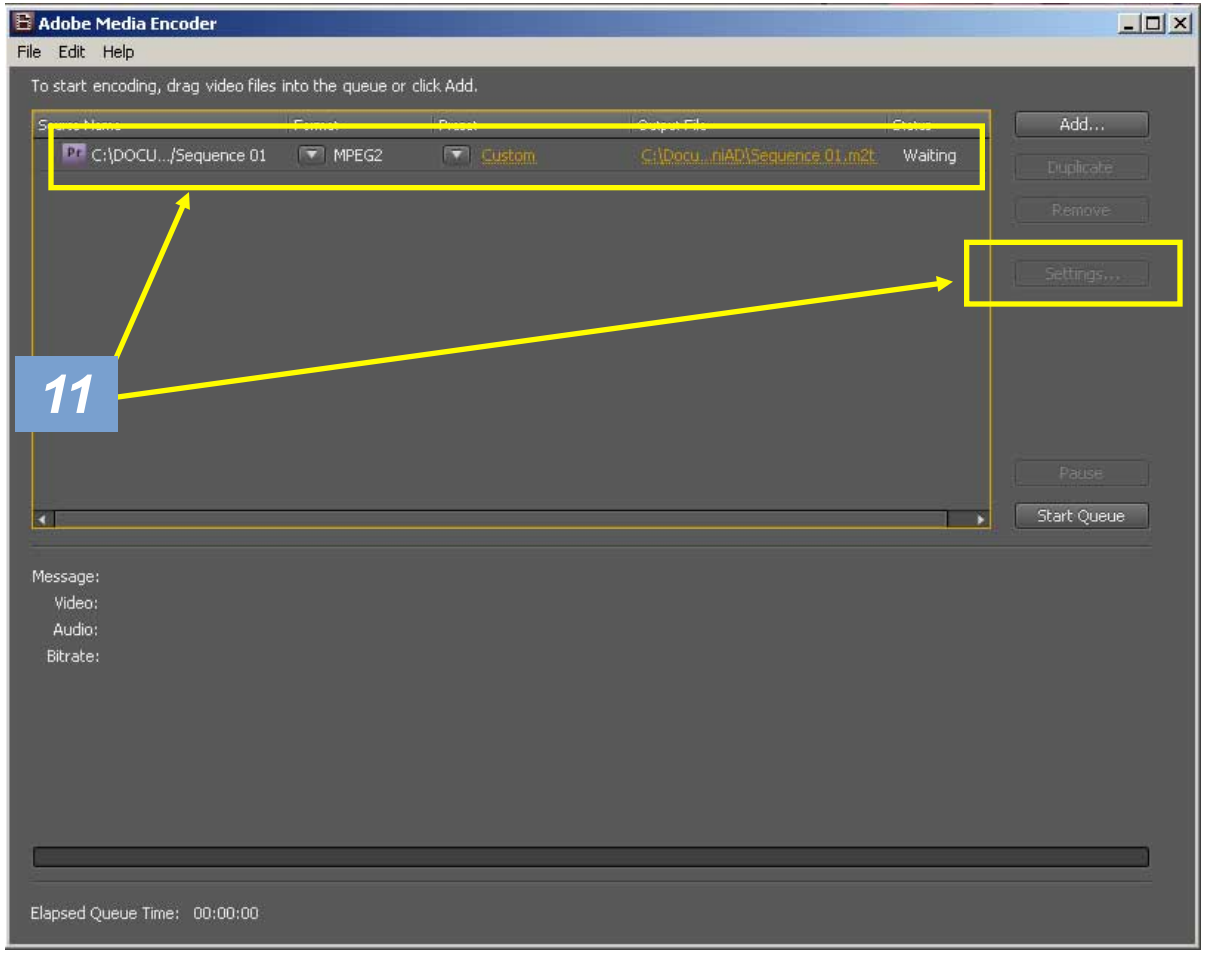


Since Adobe Media Encoder is an independent application, double-check all parameters as some settings may not transfer as expected.

Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 11*: Adobe Media Encoder**

Highlight your project in the queue and click the "Settings" button.



Since Adobe Media Encoder is an independent application, double-check all parameters.

Adobe Premiere Pro CS4: MPEG-2 Creation

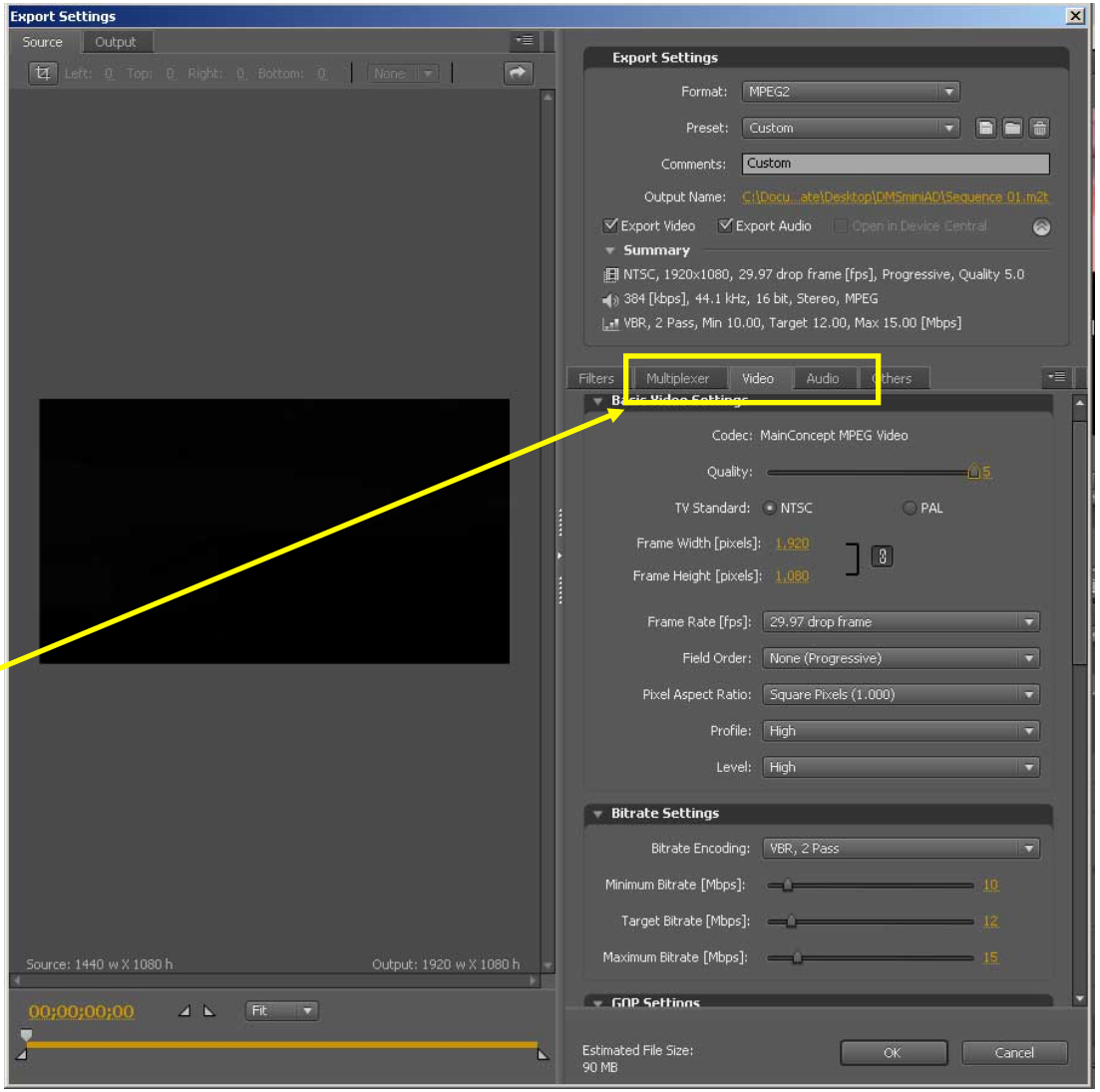
- *Step 12*: ***Adobe Media Encoder***

Since Adobe Media Encoder is an independent application, confirm all parameters for the tabs:

"Multiplexer"

"Video"

"Audio"



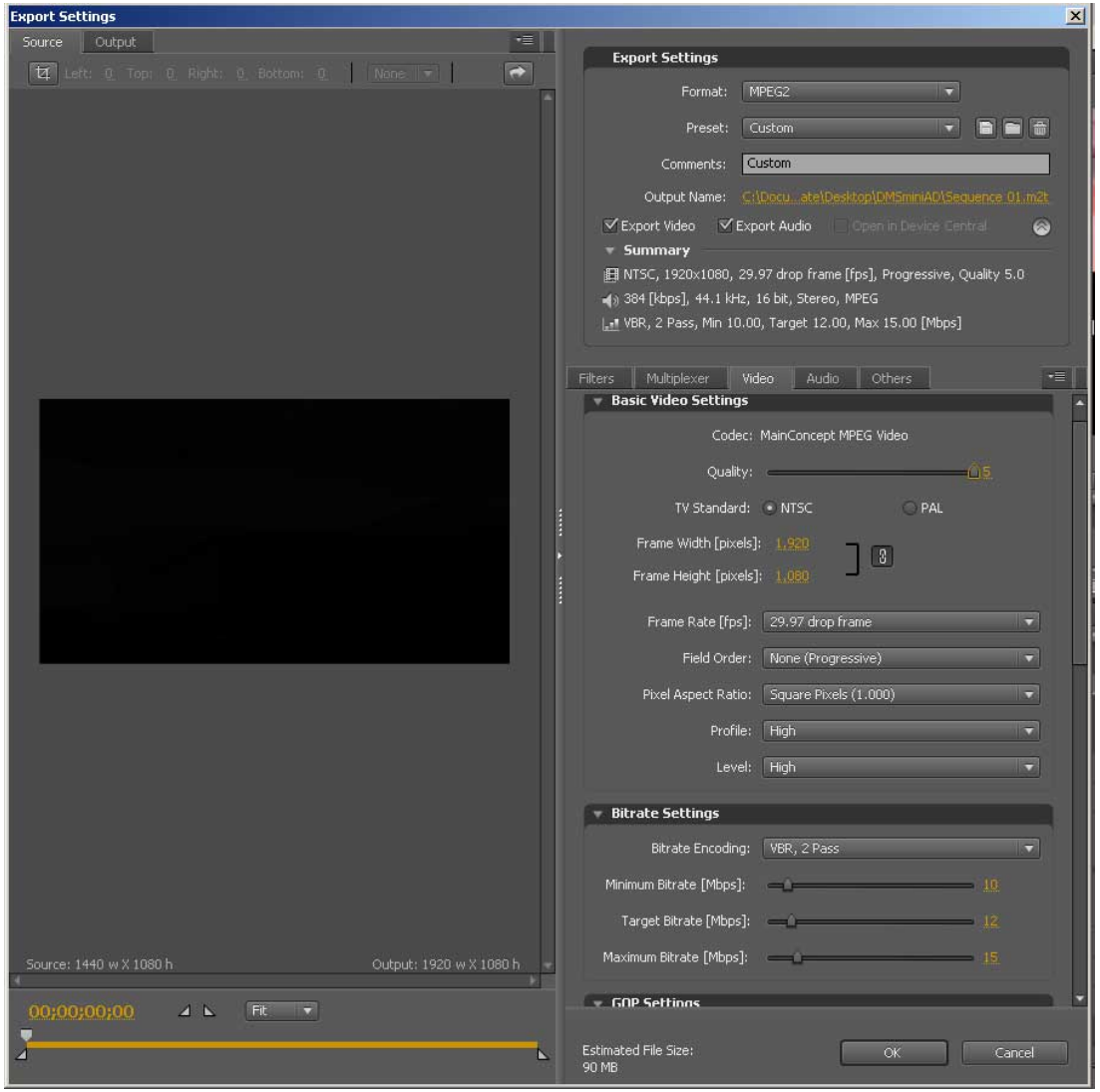
Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 13*: Adobe Media Encoder**

Audio

If the audio track is empty, include an audio export for proper transport stream function.

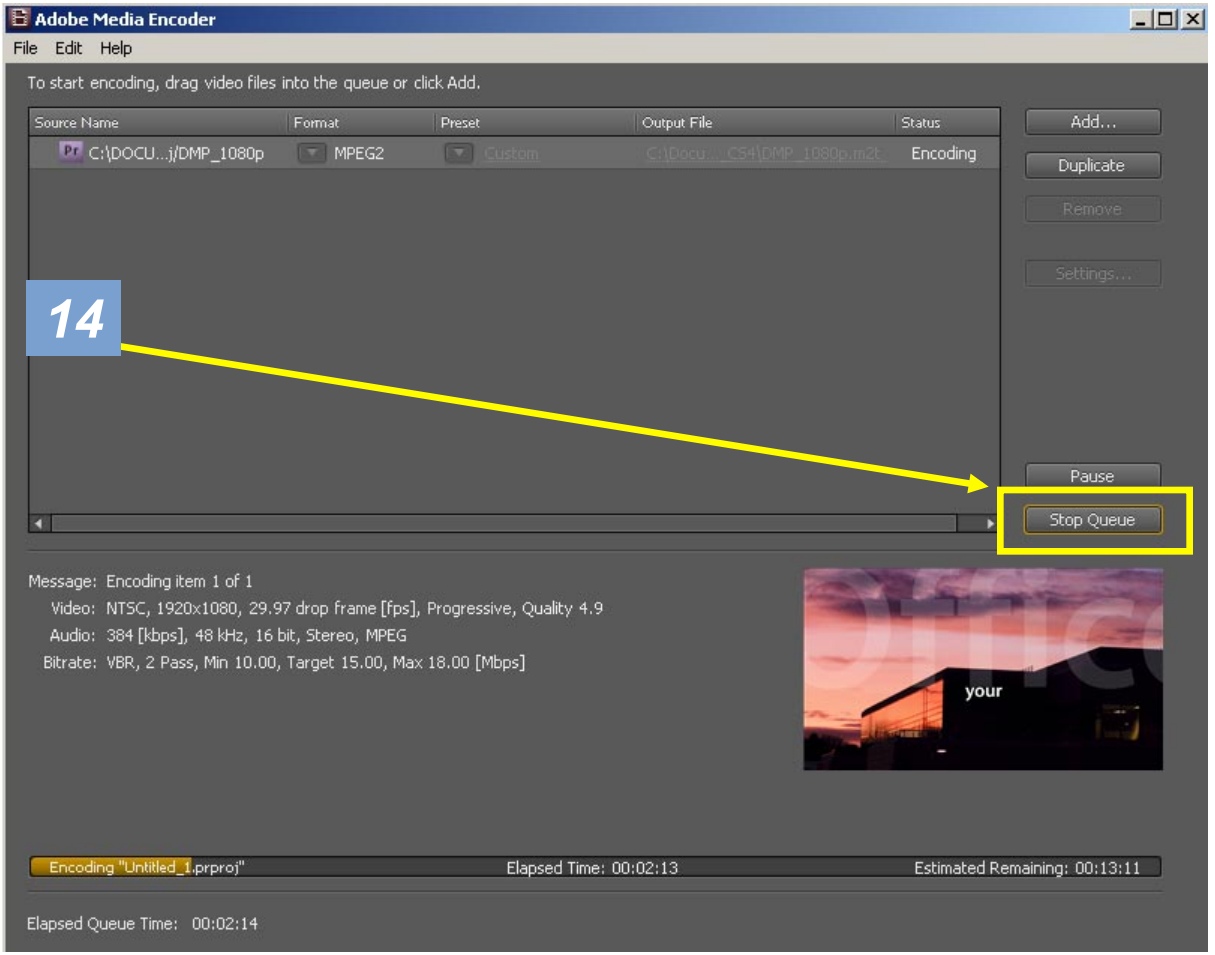
Click "***OK***"



Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 14*: Adobe Media Encoder: Start Queue**

Start Queue



Adobe Premiere Pro CS4: MPEG-2 Creation

- ***Step 15*: File extension change.**

.m2t to .mpg

Change the extension from ".m2t" to ".mpg:.

Play with your preferred player.

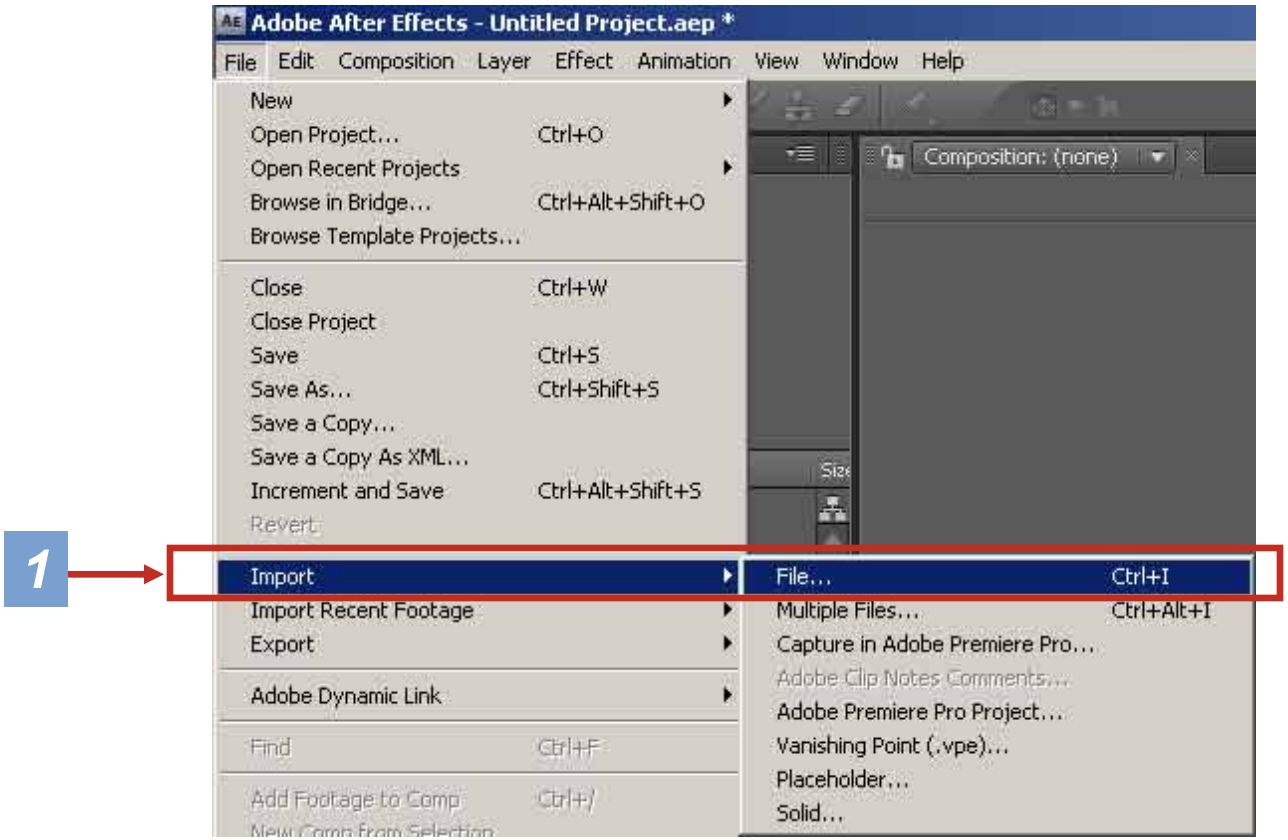
NOTE: Currently, the Quicktime MPEG add-on will not play MPEG2-TS files.

Adobe After Effects CS4: MPEG-2 TS Creation

- (Option 1) Vertical and non-standard framed videos may be saved in an uncompressed, cross-platform AVI format and used as a source to generate a custom MPEG-2 TS file in an appropriate video editing or compositing software package.
- (Option 2) Vertical and non-standard framed videos may be saved in an uncompressed, cross-platform image sequence and used as a source to generate a custom MPEG-2 TS file in an appropriate video editing or compositing software package.

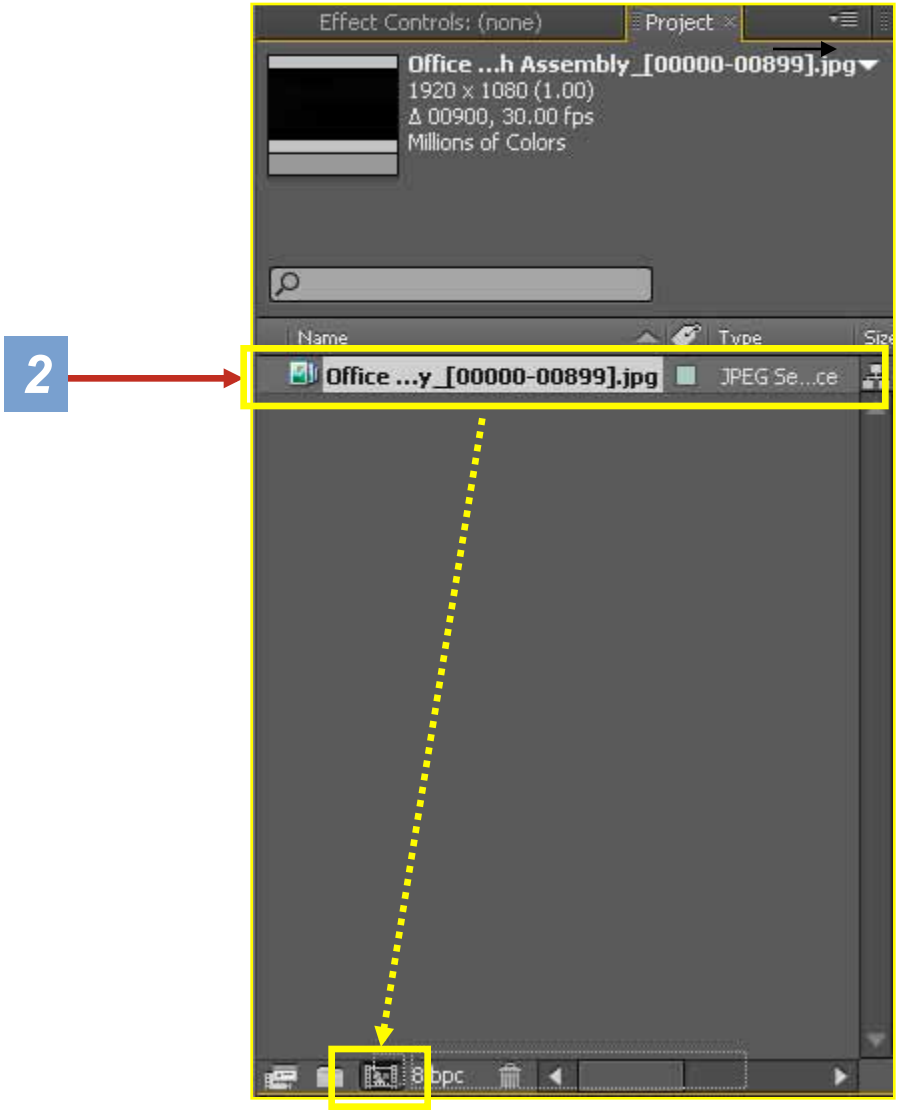
Adobe After Effects CS4: MPEG-2 TS Creation

- ***Step 1*: File → *Import file: to load footage into Project Bin***



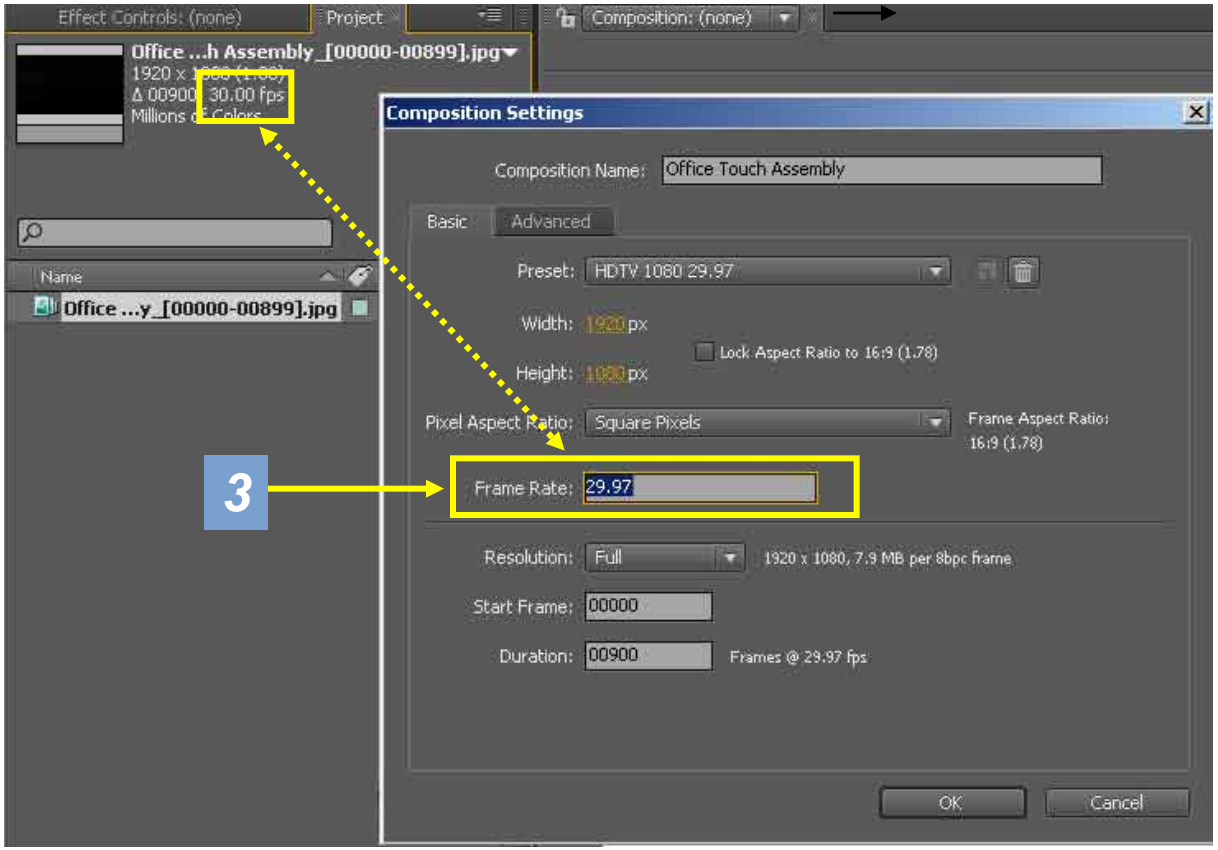
Adobe After Effects CS4: MPEG-2 TS Creation

- *Step 2*: ***Create*** your composite by dragging the file onto comp button



Adobe After Effects CS4: MPEG-2 TS Creation

- ***Step 3:*** ***Conform both footage and composition frame rates.***

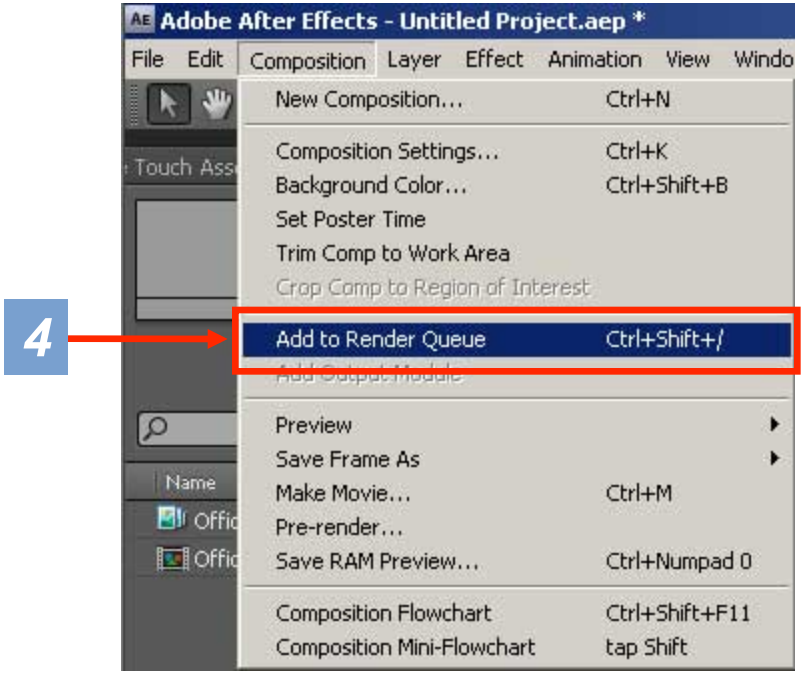


29.97fps adjustment for:

Footage: File>Interpret Footage>Main…
Composition: Composition>Composition Settings…

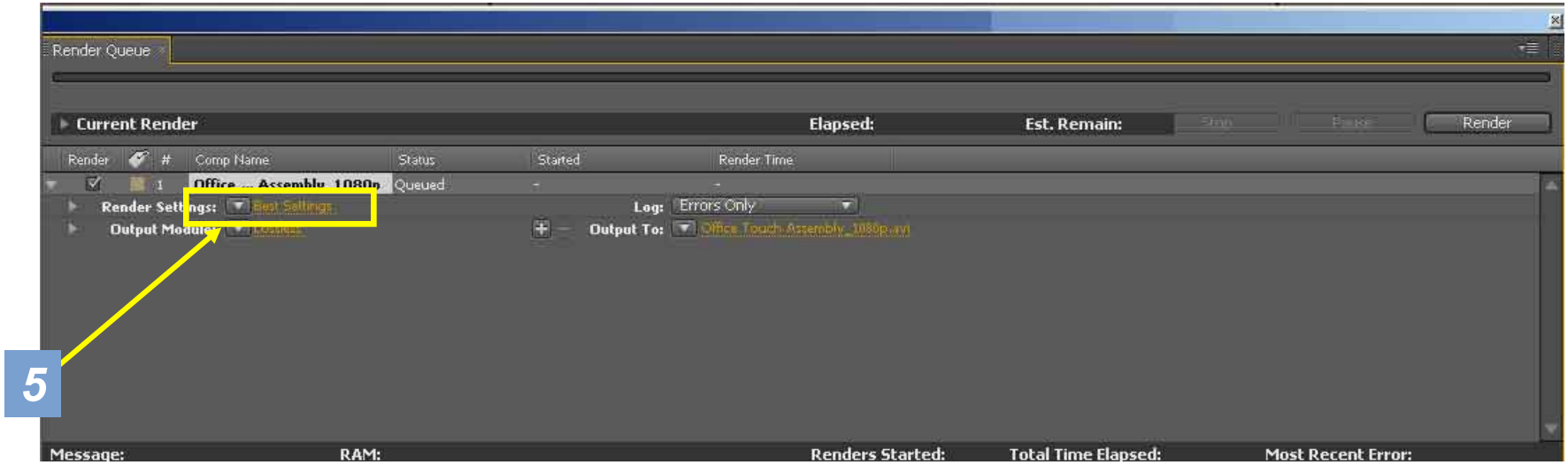
Adobe After Effects CS4: MPEG-2 TS Creation

- *Step 4*: ***Render queue***: Composition > ”Add to Render Queue”



Adobe After Effects CS4: MPEG-2 TS Creation

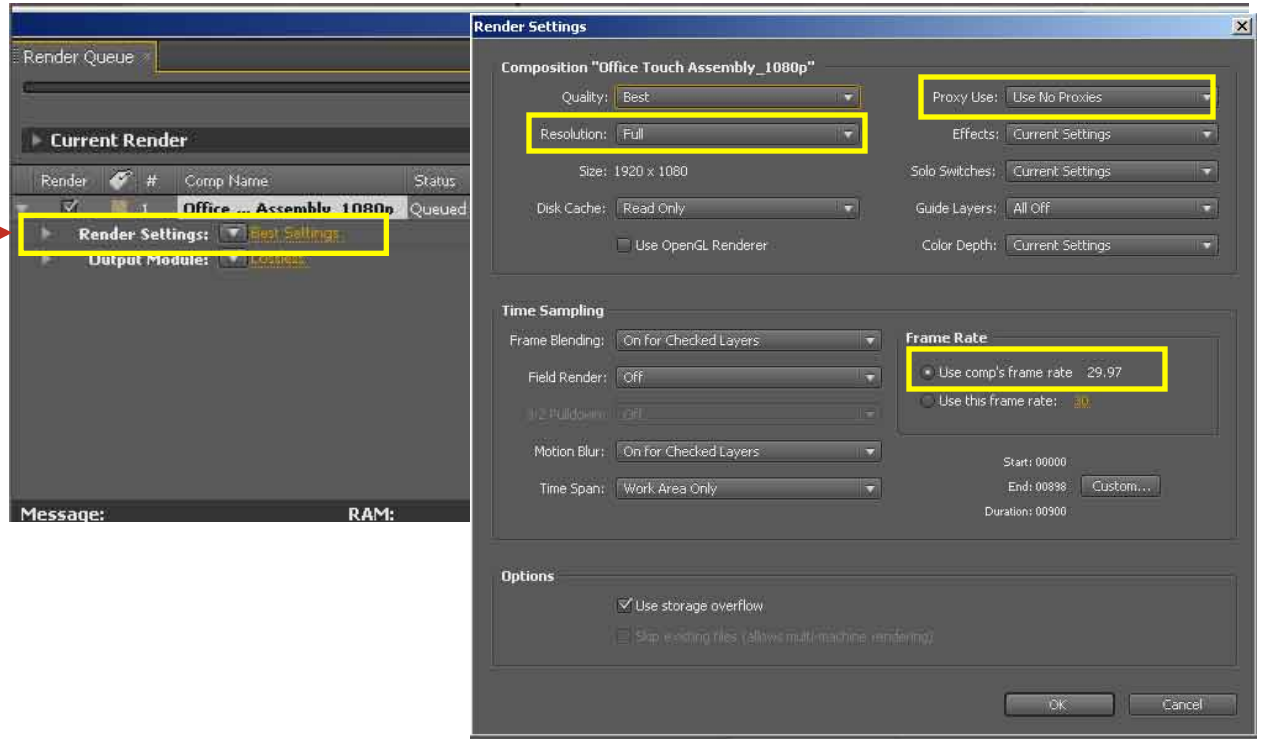
- ***Step 5:*** *Launch* the Render Settings Module by clicking “Best Settings”



Adobe After Effects CS4: MPEG-2 TS Creation

- ***Step 5a*: Check that settings match.**

5a



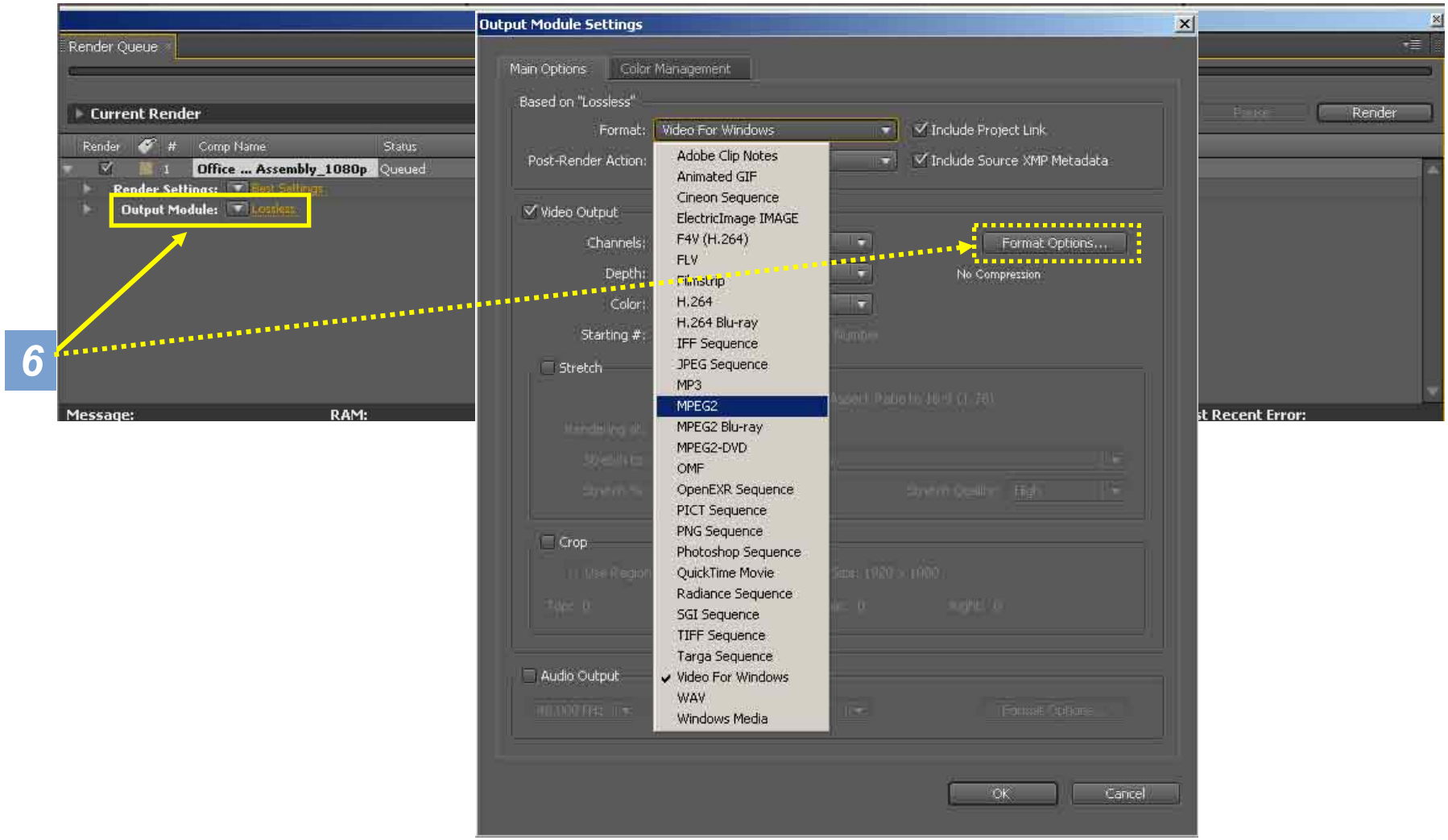
Verify:

- ✓ *Resolution = Full*
- ✓ *Proxy Use = No Proxies*
- ✓ *Use comp frame rate*
- ✓ *Click "OK"*

Adobe After Effects CS4: MPEG-2 TS Creation

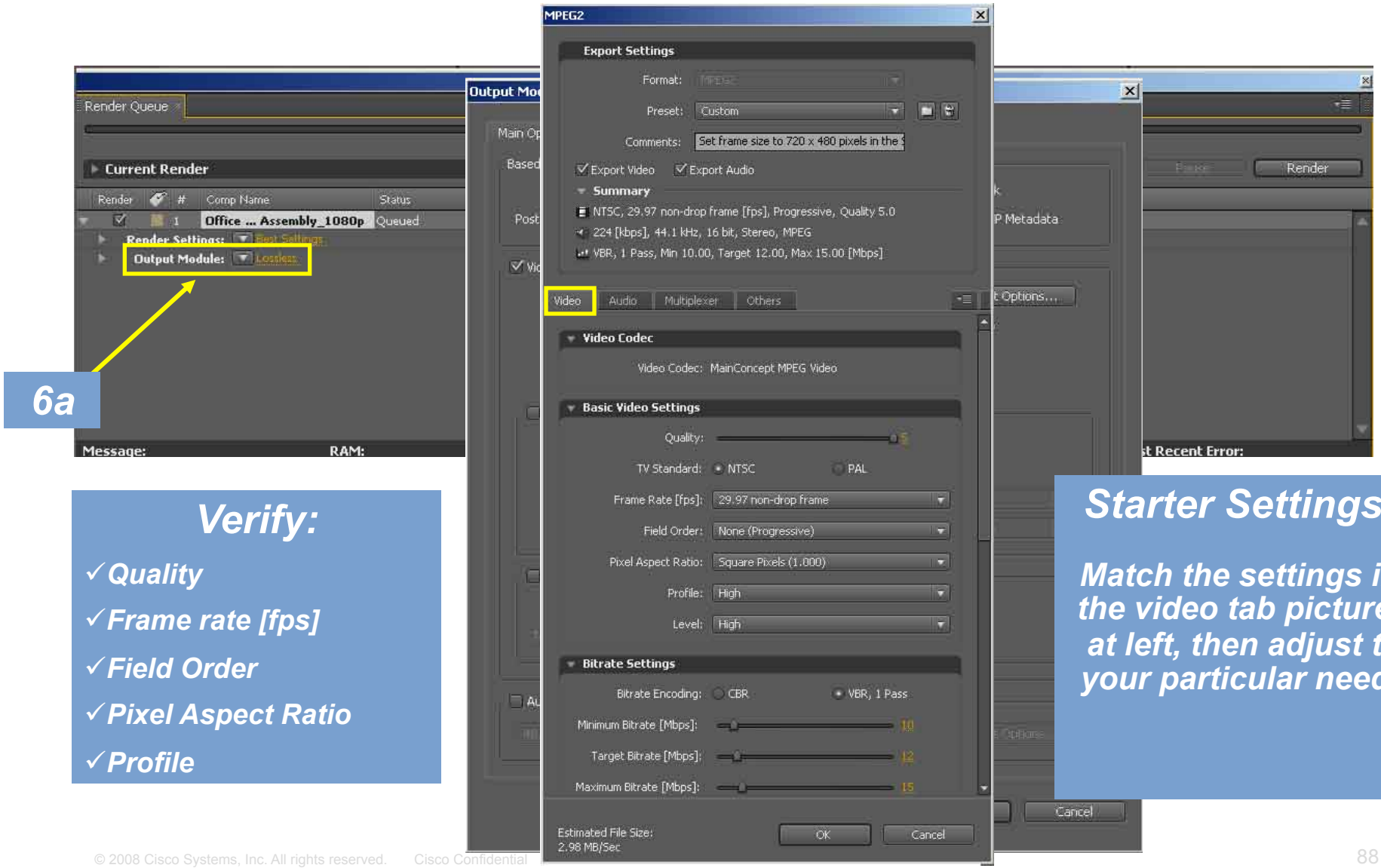
- ***Step 6*: *Launch*** the Output Module by clicking “Lossless”

  Select: Format > MPEG2. Then “Format Options...”



Adobe After Effects CS4: MPEG-2 TS Creation

- ***Step 6a*: *Select* : The "Video" tab.**



Verify:

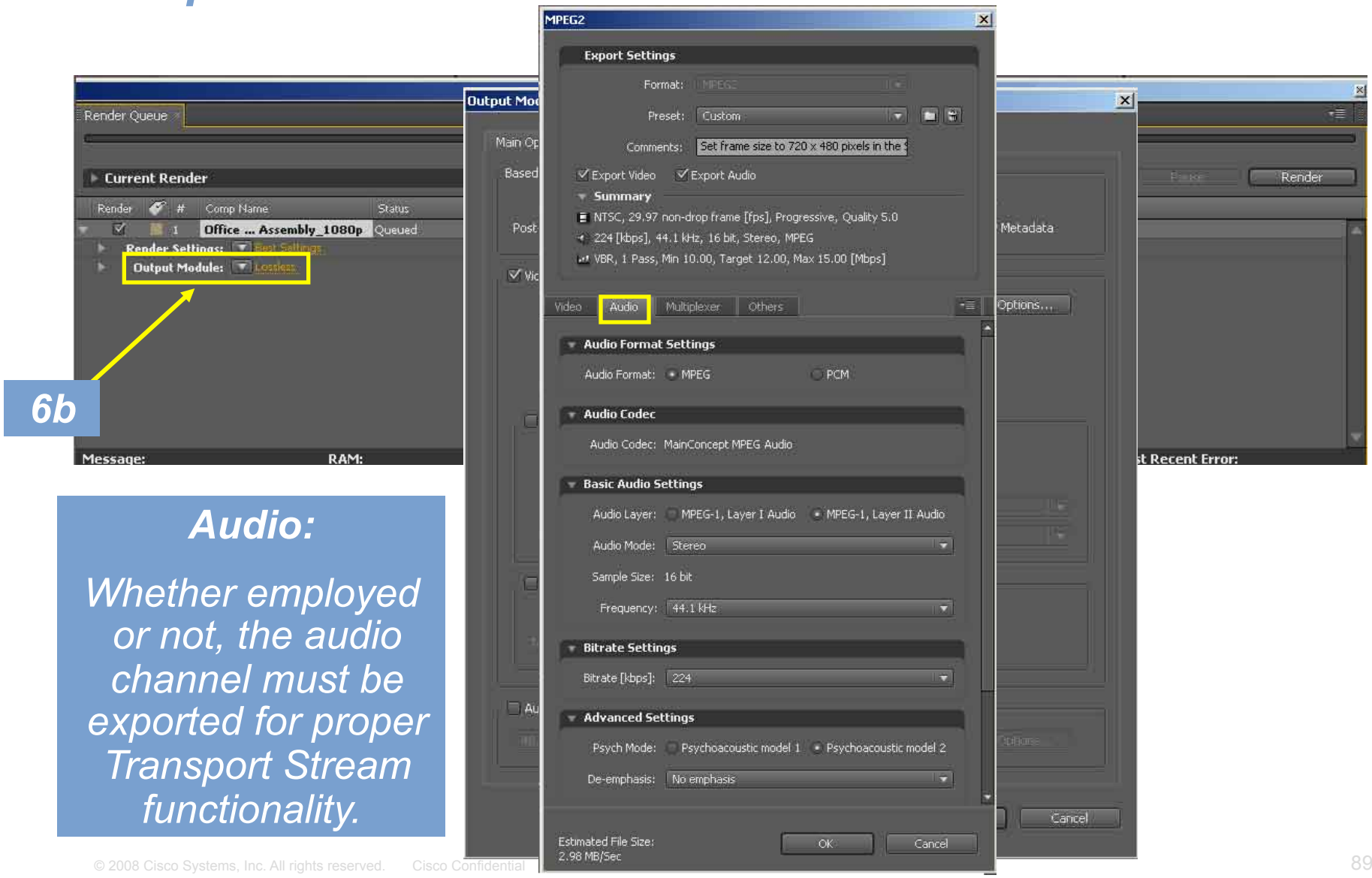
- ✓ *Quality*
- ✓ *Frame rate [fps]*
- ✓ *Field Order*
- ✓ *Pixel Aspect Ratio*
- ✓ *Profile*

Starter Settings:

Match the settings in the video tab pictured at left, then adjust to your particular needs

Adobe After Effects CS4: MPEG-2 TS Creation

- ***Step 6b*: *Select*** : The “Audio” tab.

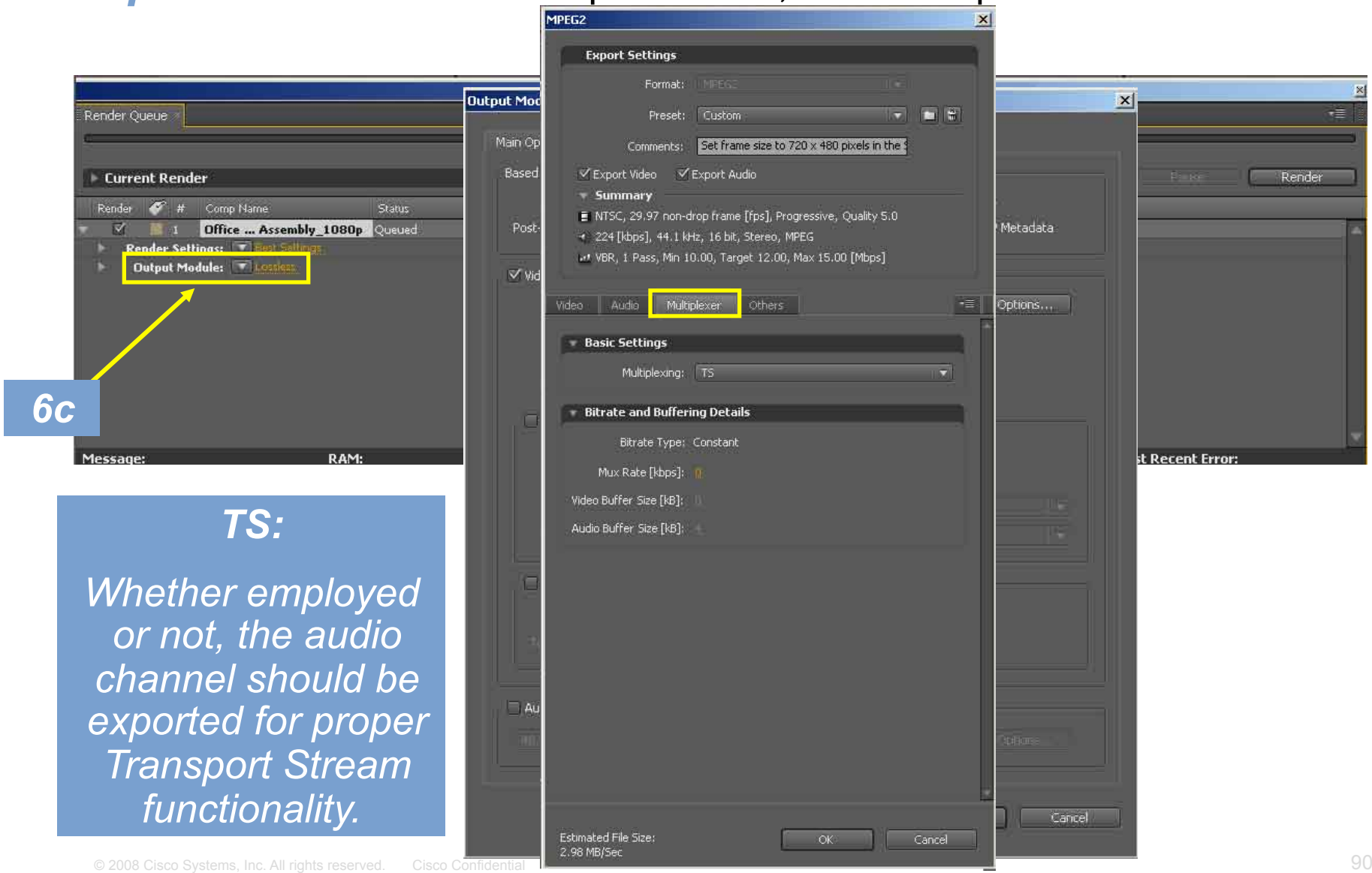


Audio:

Whether employed or not, the audio channel must be exported for proper Transport Stream functionality.

Adobe After Effects CS4: MPEG-2 TS Creation

- ***Step 6c*: *Select*** : The “Multiplexer” tab, and “Multiplexer =TS”. “OK”

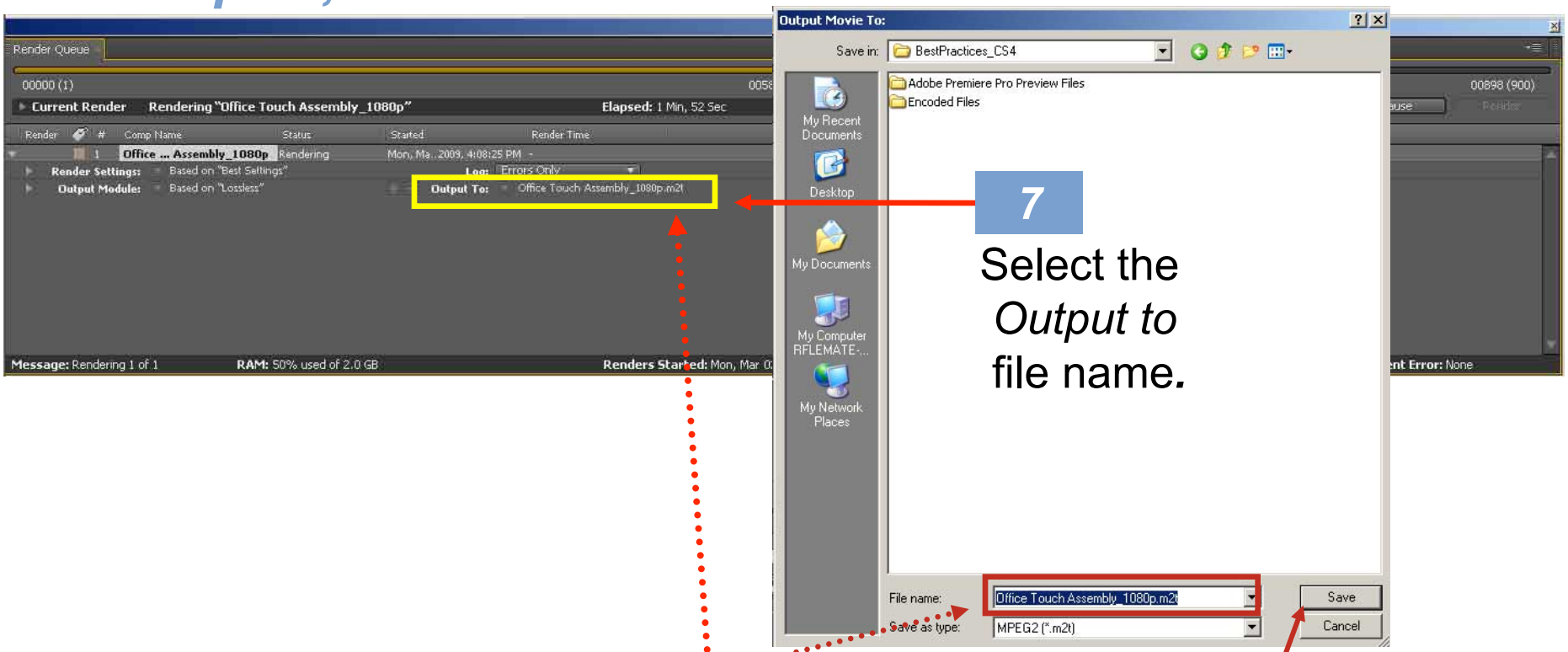


TS:

Whether employed or not, the audio channel should be exported for proper Transport Stream functionality.

Adobe After Effects CS4: MPEG-2 TS Creation

- *Steps 7, 8:*



7

Select the *Output to* file name.

8

Click, ***Save.***

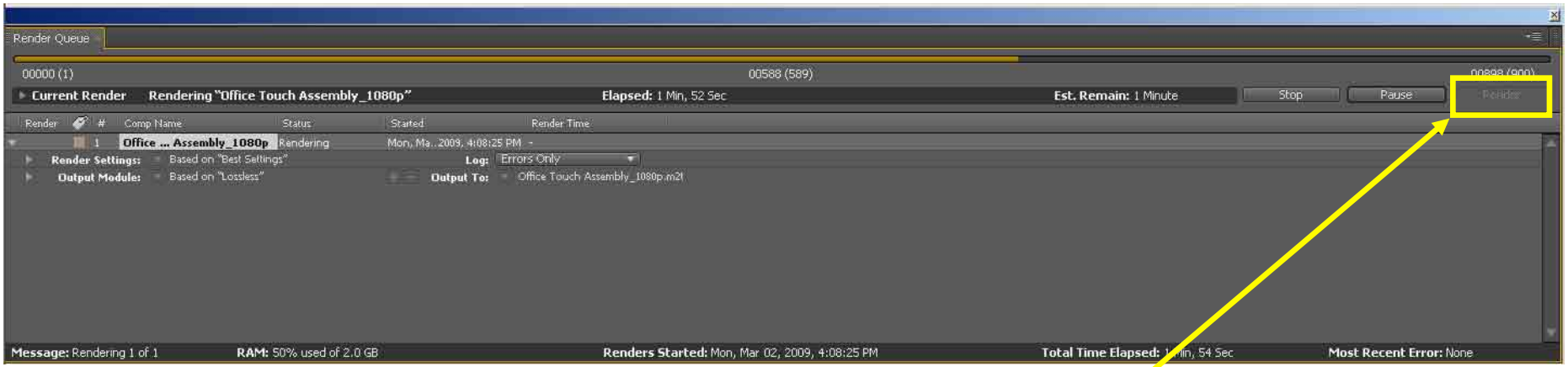
Note: *the file type: .m2t = MPEG2-TS*

Change to ".mpg" for

Note: This "Save" means "Save file to" and does not start the rendering

Adobe After Effects CS4: MPEG-2 TS Creation

- ***Steps 9: Generating your MPEG2-TS file.***



- ***Step 9:*** Click, ***Render.*** ***to output the MPEG2-TS file.***

- ***(optional) :*** To save on file size you may also use the VLC player to transcode the "mp2v" codec to an h.264 codec. ***See "MPEG2 in VLC"above.***

Adobe After Effects CS4: MPEG-2 TS Creation

- ***Step 10*: File extension change.**

.m2t to .mpg

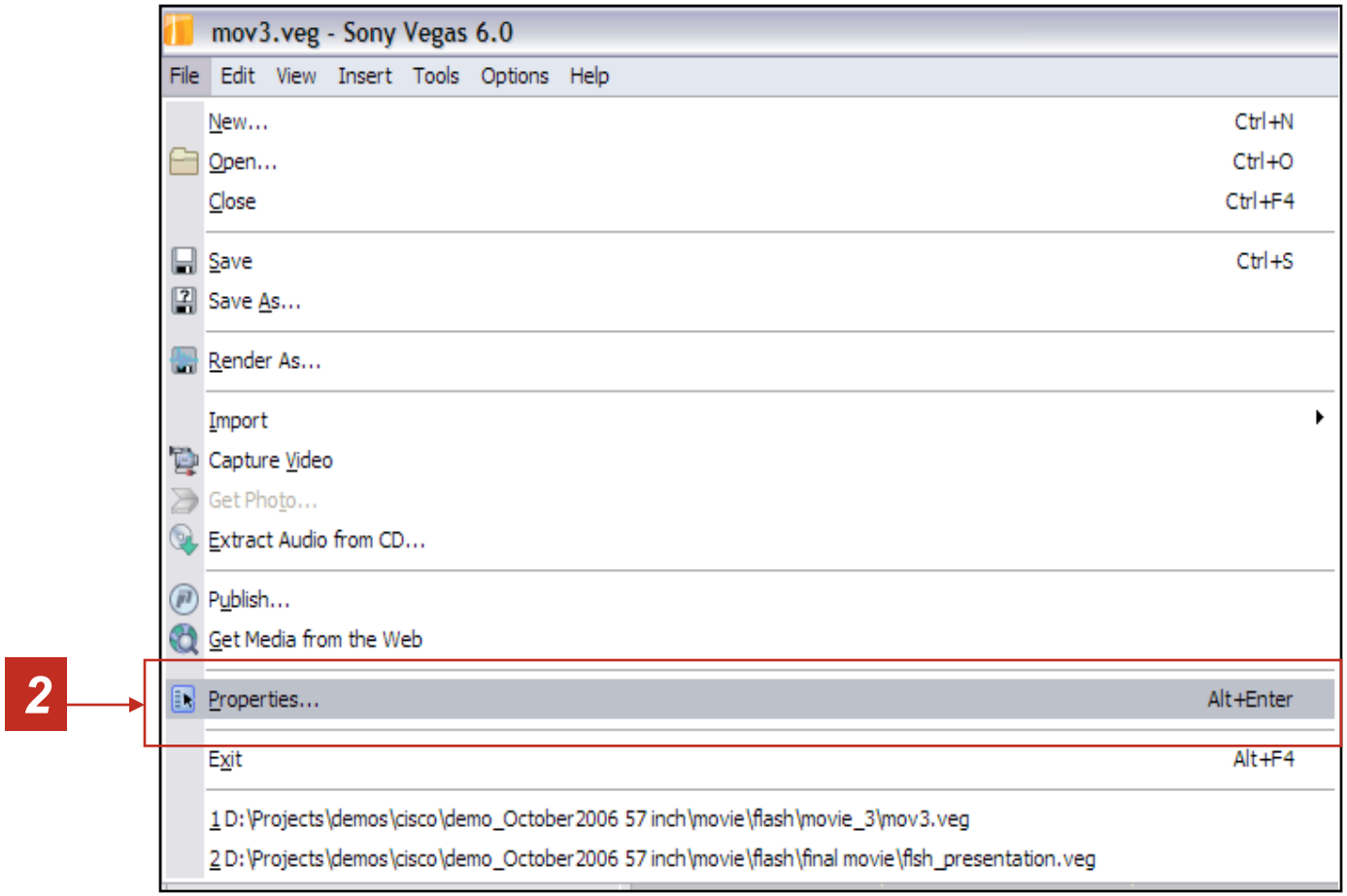
Change the extension from ".m2t" to ".mpg:.

Play with your preferred player.

NOTE: Currently, the Quicktime MPEG add-on will not play MPEG2-TS files.

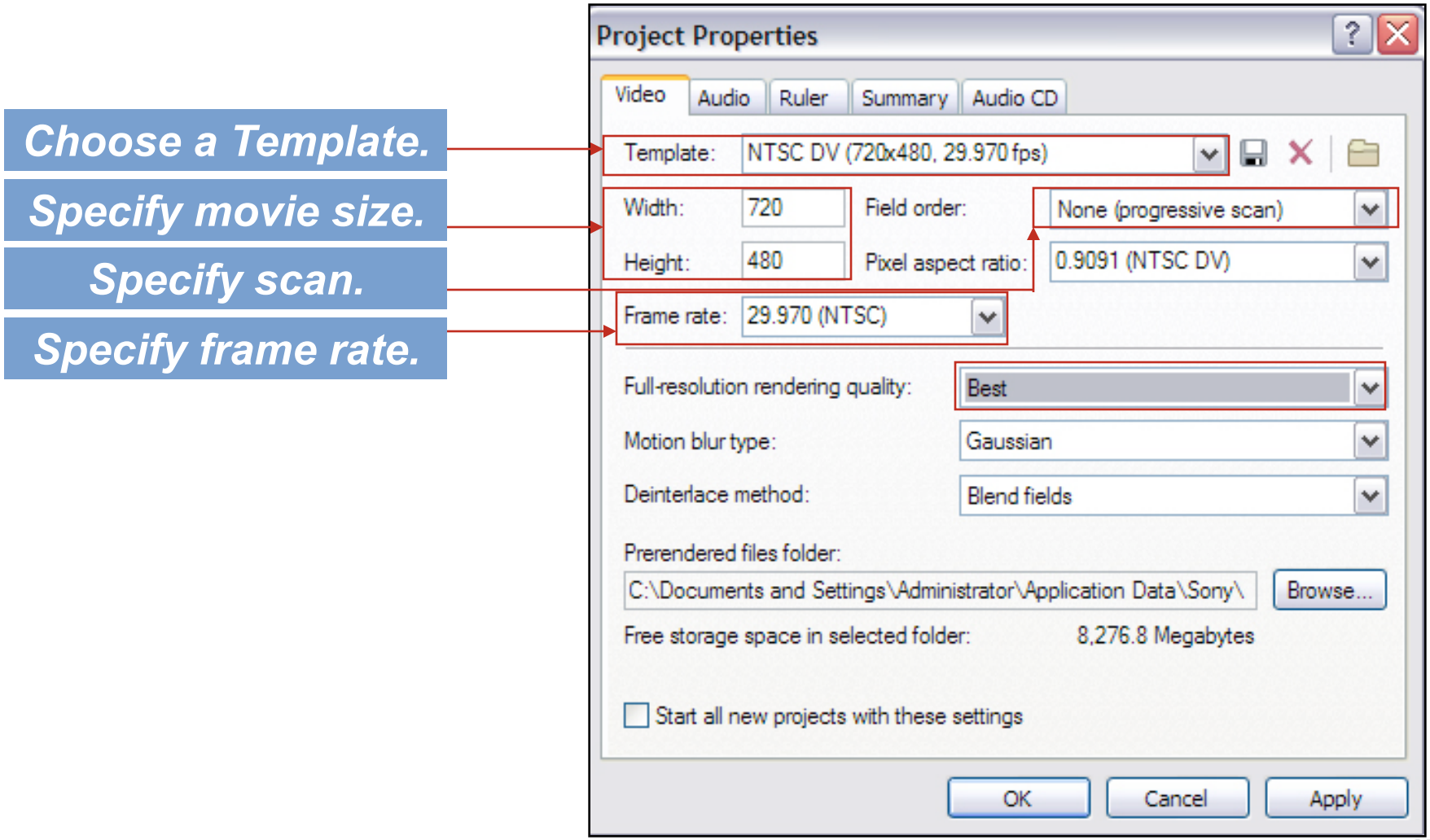
Sony Vegas: MPEG-2 Rendering

- ***Step 1:*** Import the file you need to encode into MPEG2
- ***Step 2:*** Choose ***File → Properties***



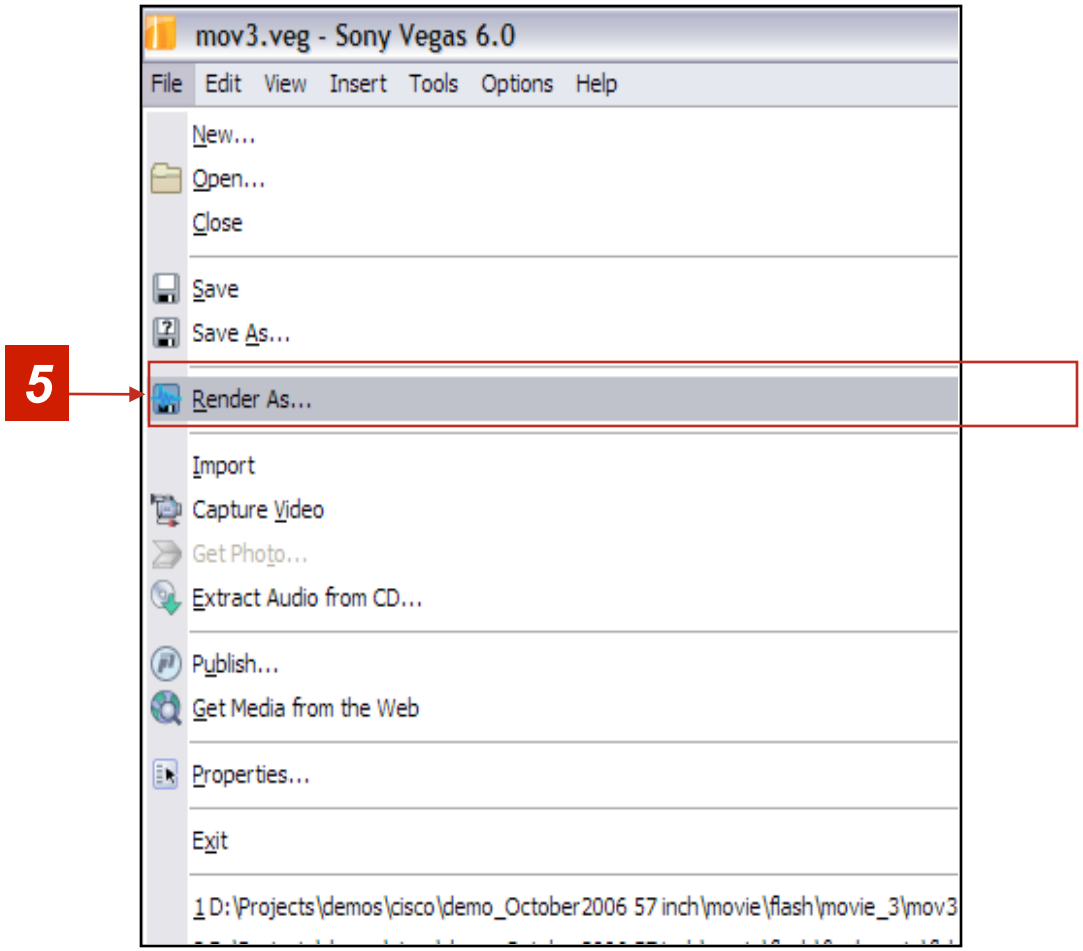
Sony Vegas: MPEG-2 Rendering

- ***Step 3:*** Choose properties: choose a template you want to use from the drop-down menu (NTSC DV may be replaced with HDV 720-30p for example); specify width, height, etc.



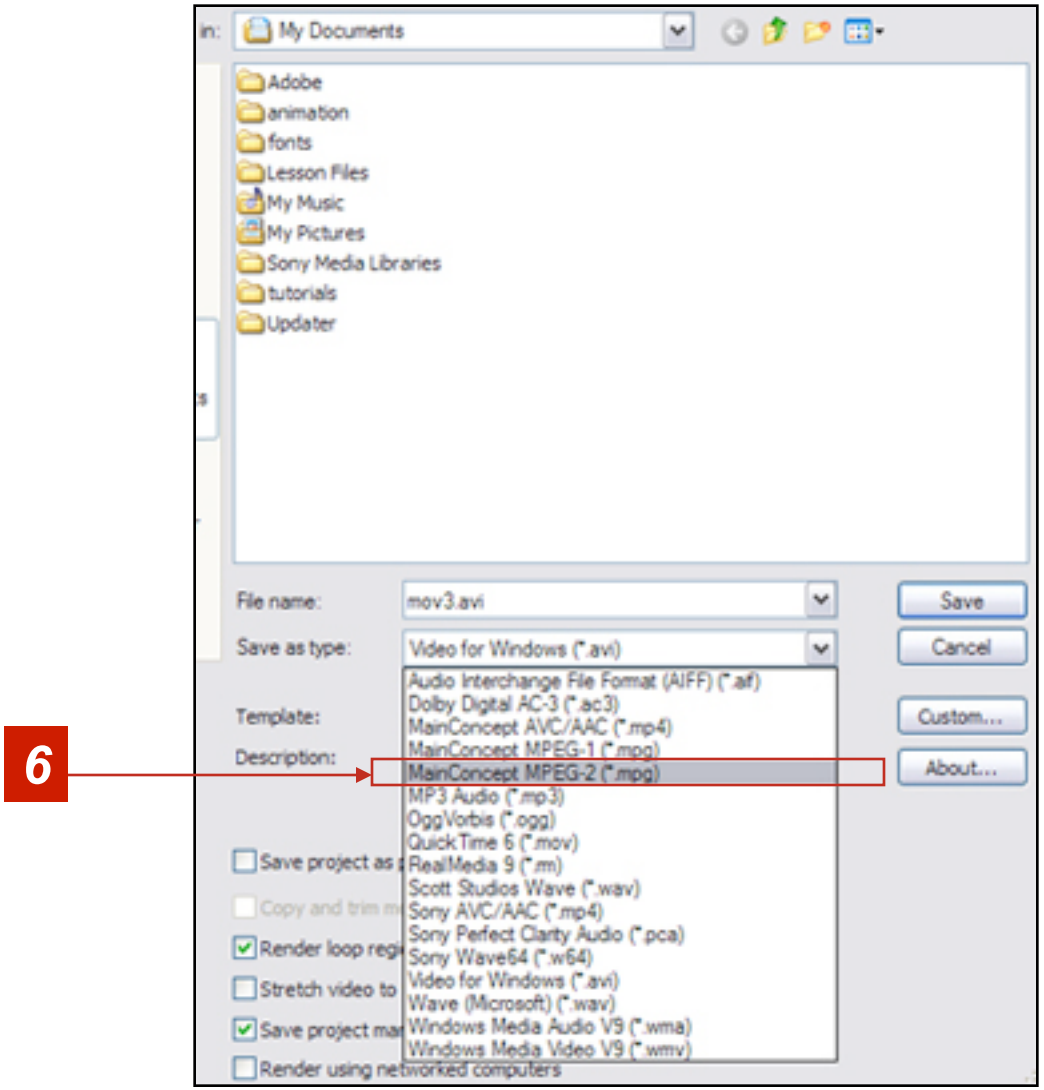
Sony Vegas: MPEG-2 Rendering

- ***Step 4:*** Create your movie
- ***Step 5:*** From ***file*** menu choose "***render as***" to save the movie



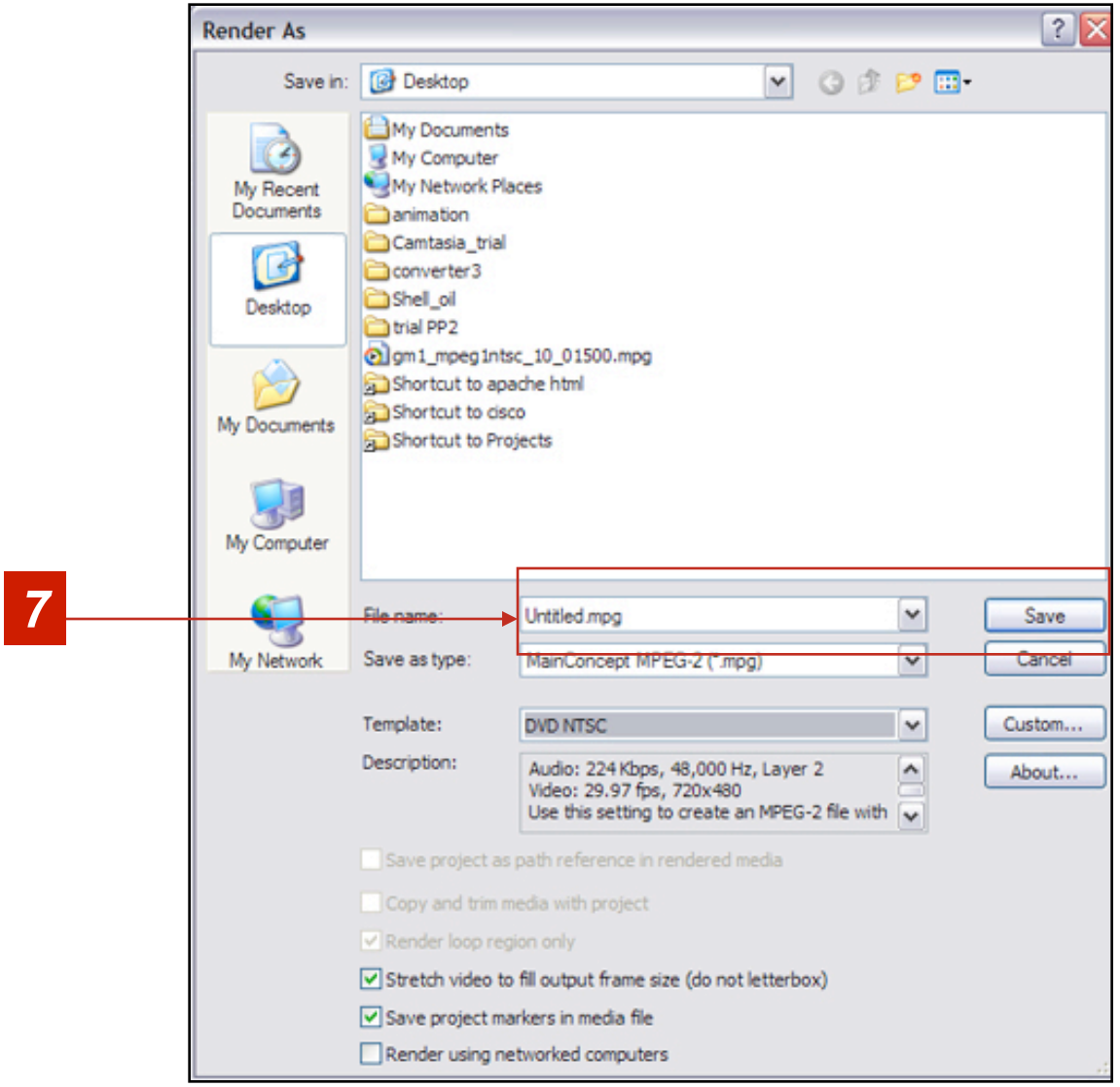
Sony Vegas: MPEG-2 Rendering

- ***Step 6:*** Choose ***MainConcept MPEG2*** from the drop-down menu



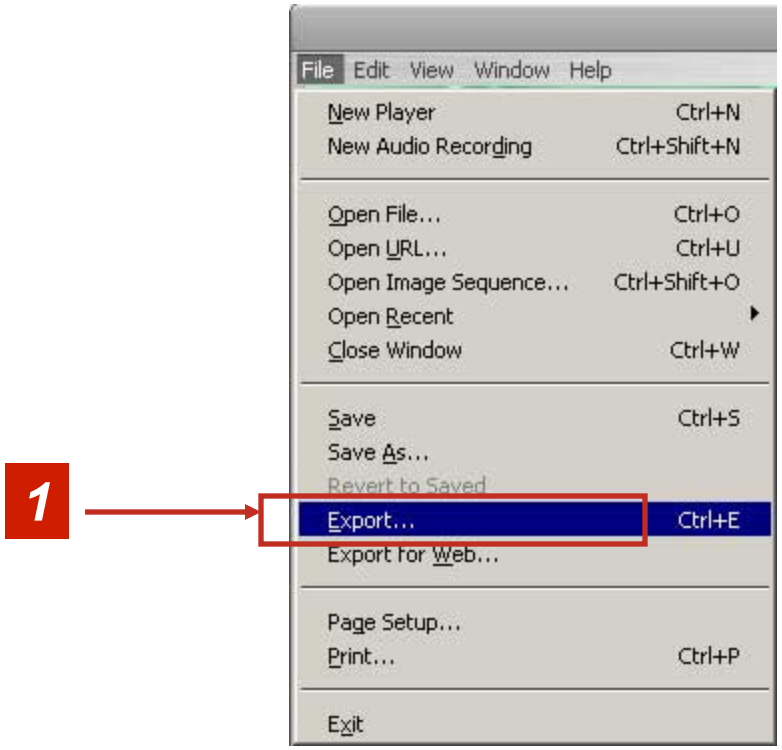
Sony Vegas: MPEG-2 Rendering

- *Step 7:* **Name the video file and save**



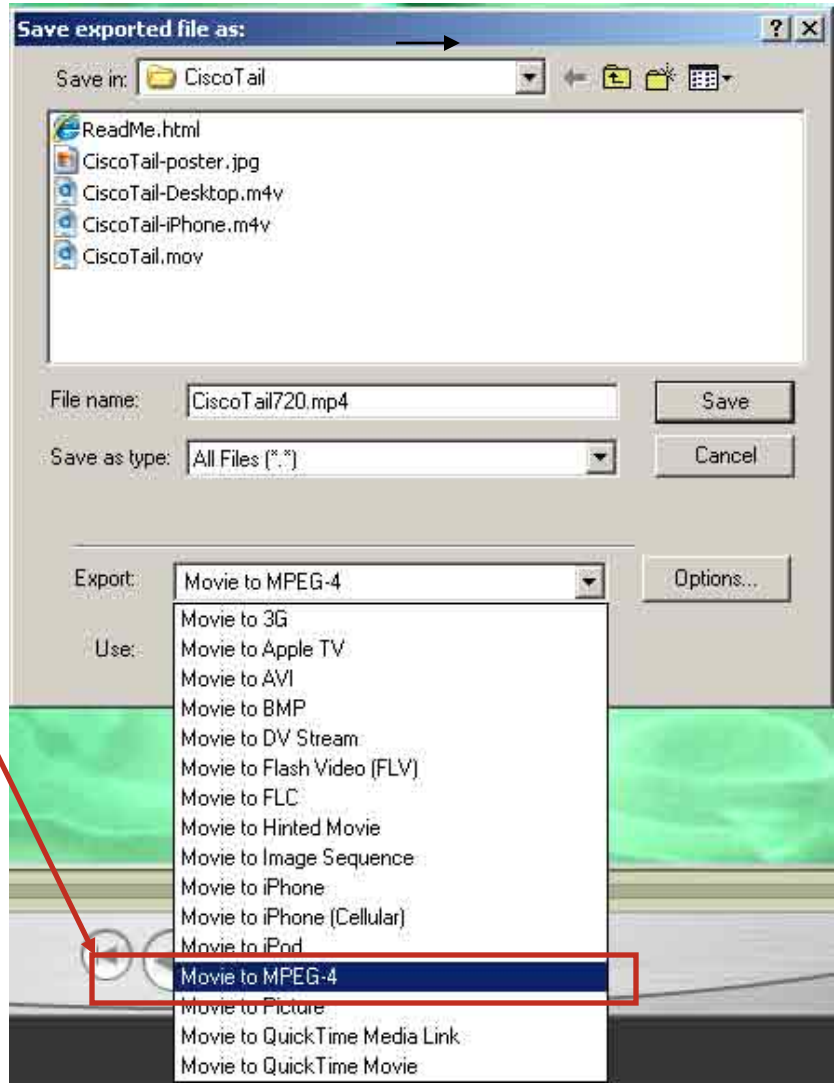
Quicktime Pro: MPEG-4 Export

***Step 1*: After opening a desired video file, select File>Export…**



Quicktime Pro: MPEG-4 Export

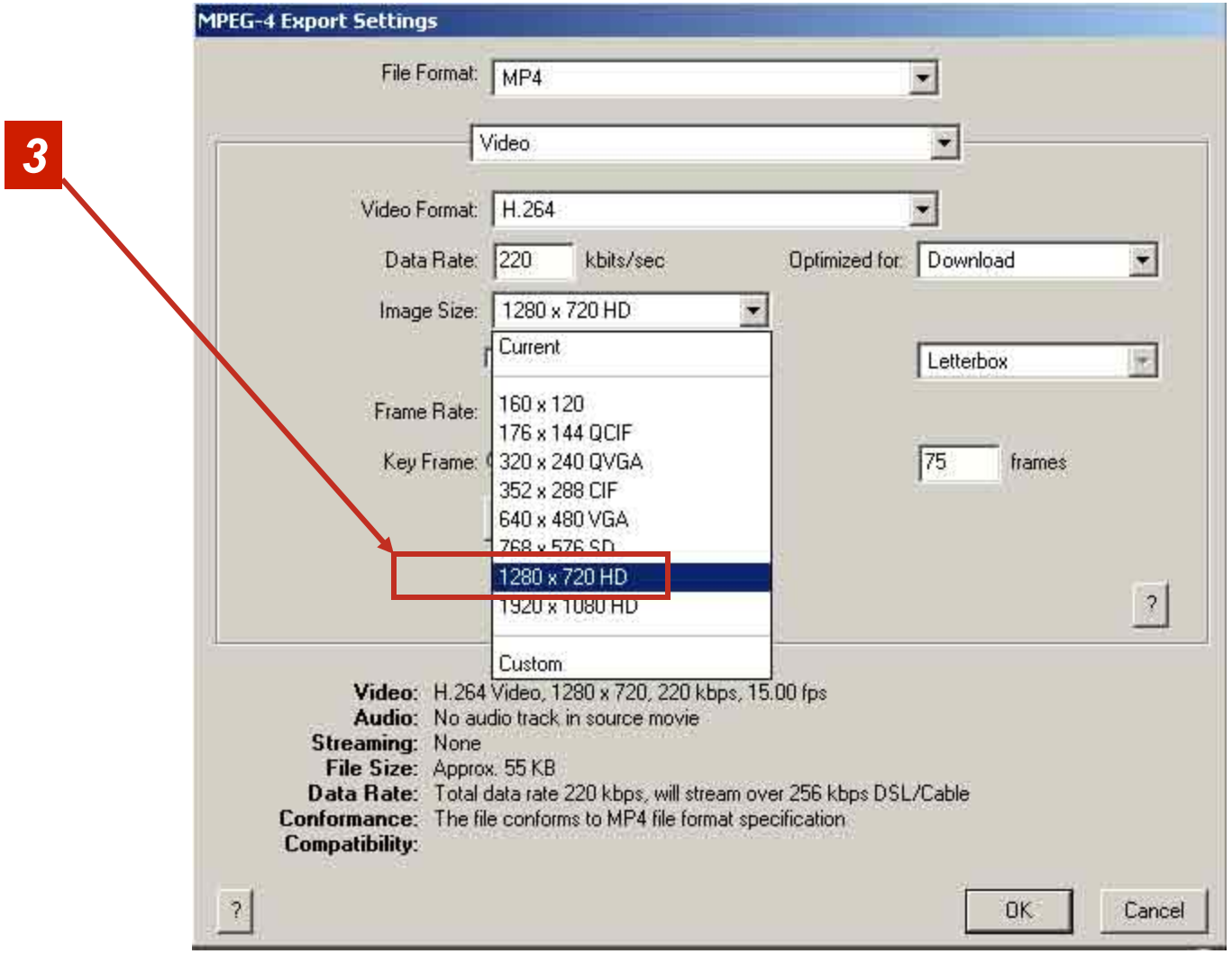
***Step 2*: From the Export menu select: "Movie to MPEG-4"**



***Note*: The use of the VLC player to wrap the .mp4v file into the MPEG2-TS format will be necessary; however, transcoding is not a necessary procedure.**

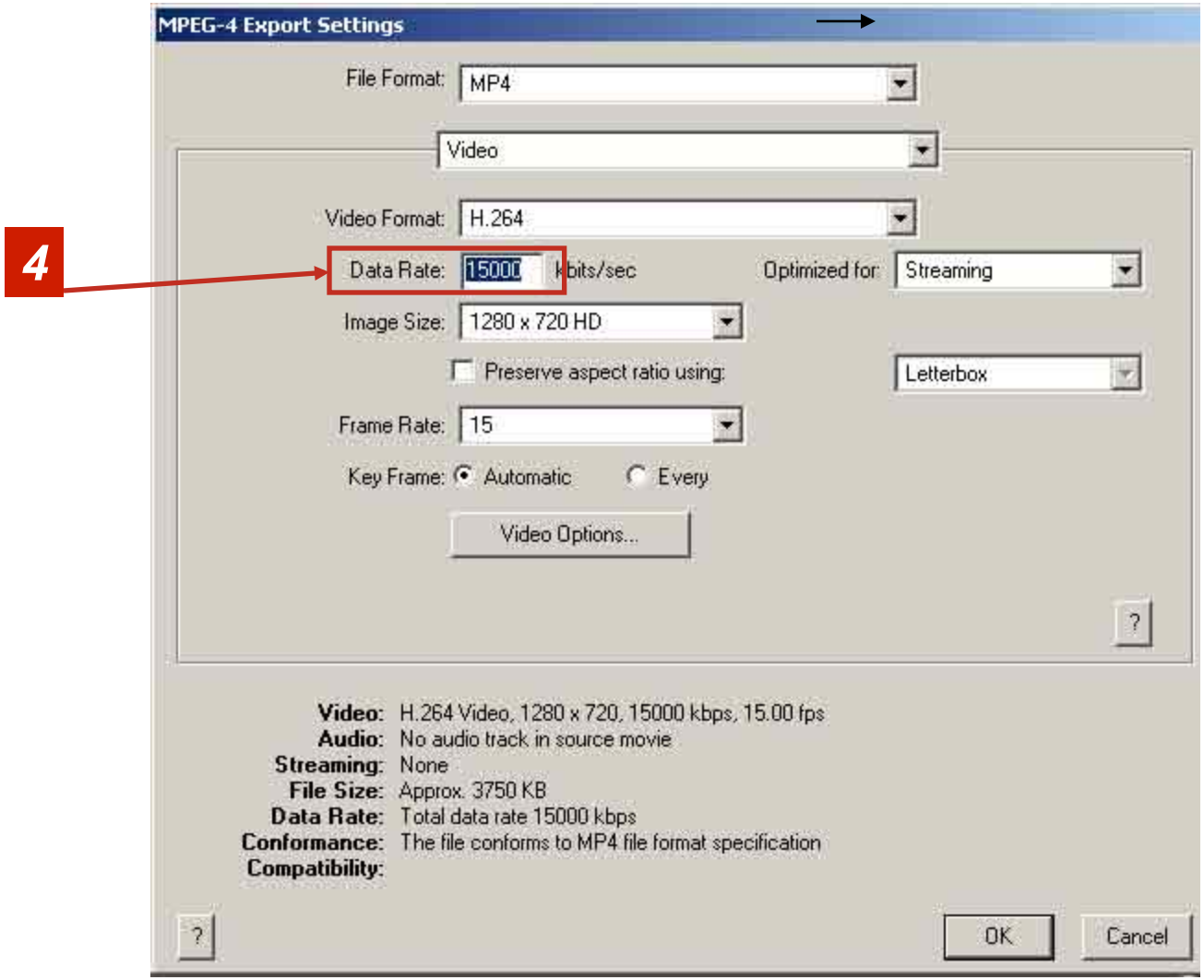
Quicktime Pro: MPEG-4 Export

Step 3: Set Image Size to 720p or 1080p.



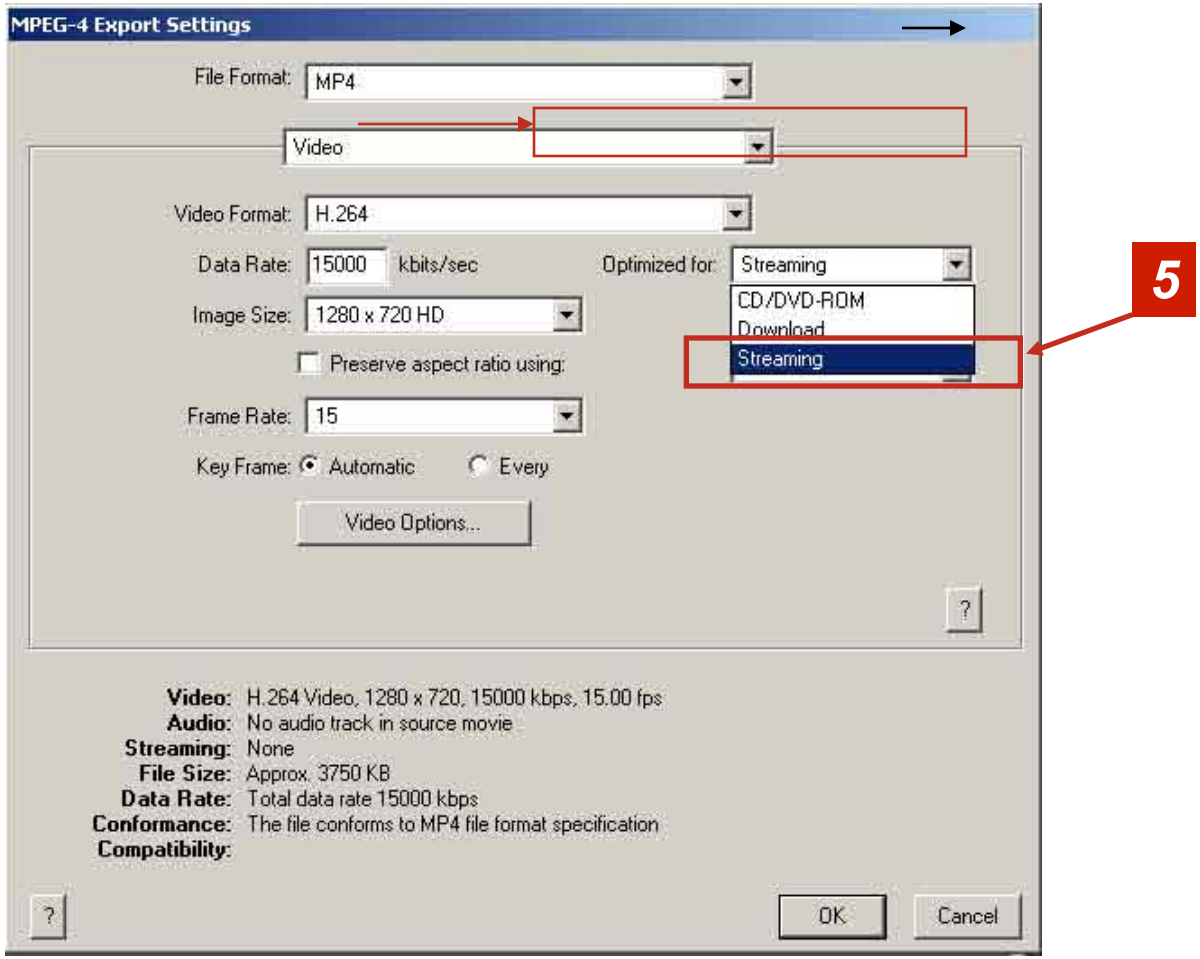
Quicktime Pro: MPEG-4 Export

Step 4: ***Select an appropriate data rate from 12 to 15Mbps.***



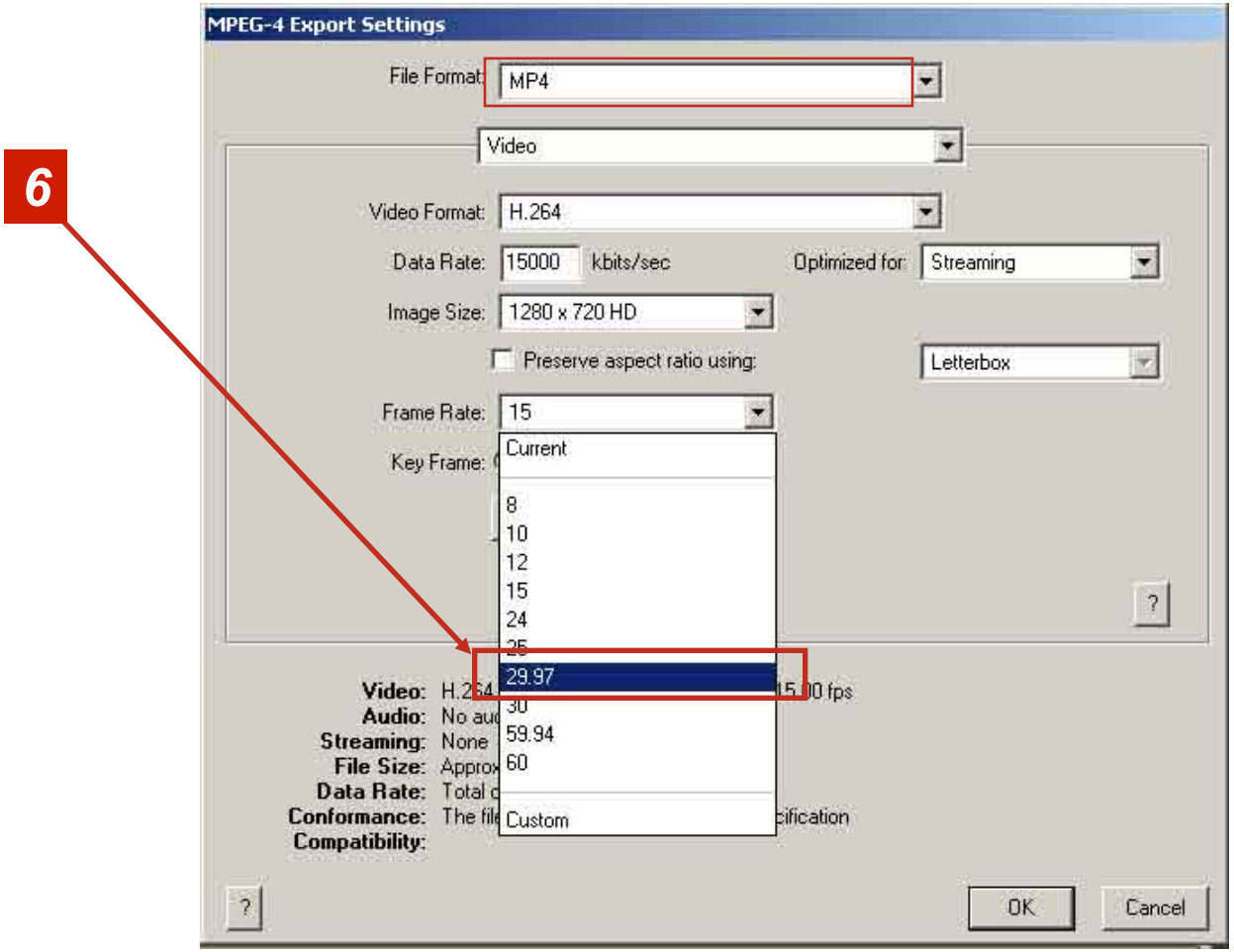
Quicktime Pro: MPEG-4 Export

Step 5: ***Select an appropriate data rate from 12 to 15Mbps.***



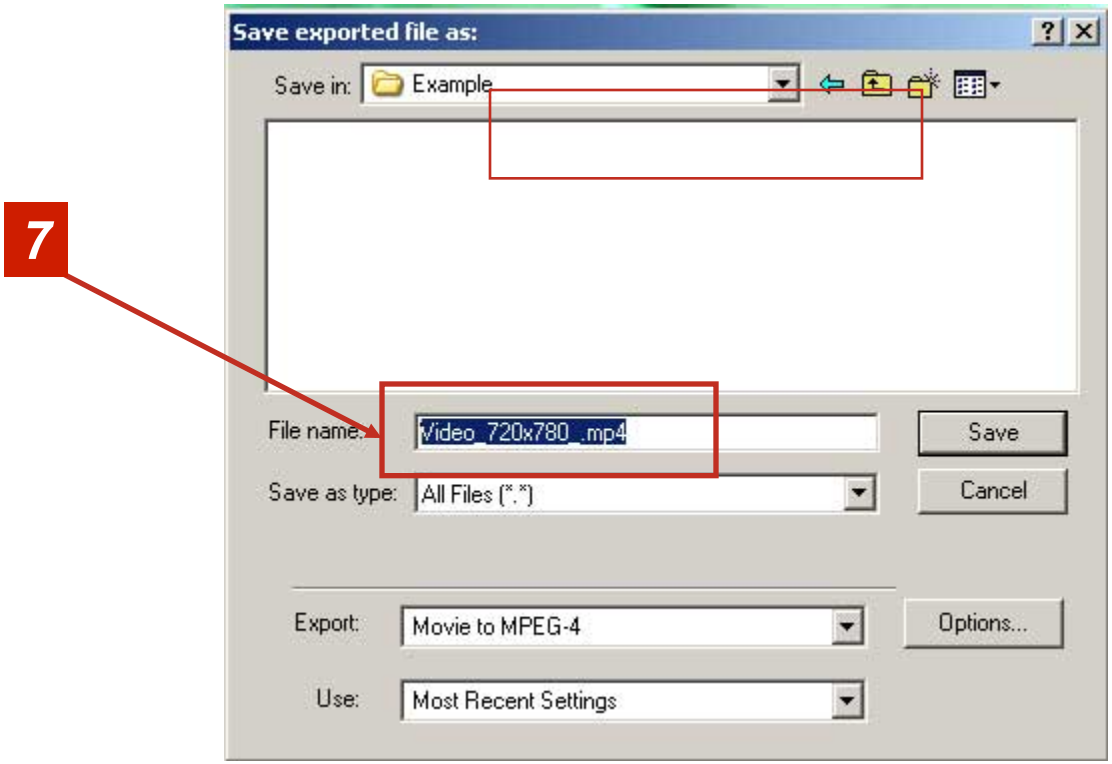
Quicktime Pro: MPEG-4 Export

Step 6: ***Select an appropriate video frame rate.***



Quicktime Pro: MPEG-4 Export

Step 7: ***Enter*** **a filename, then click "Save".**

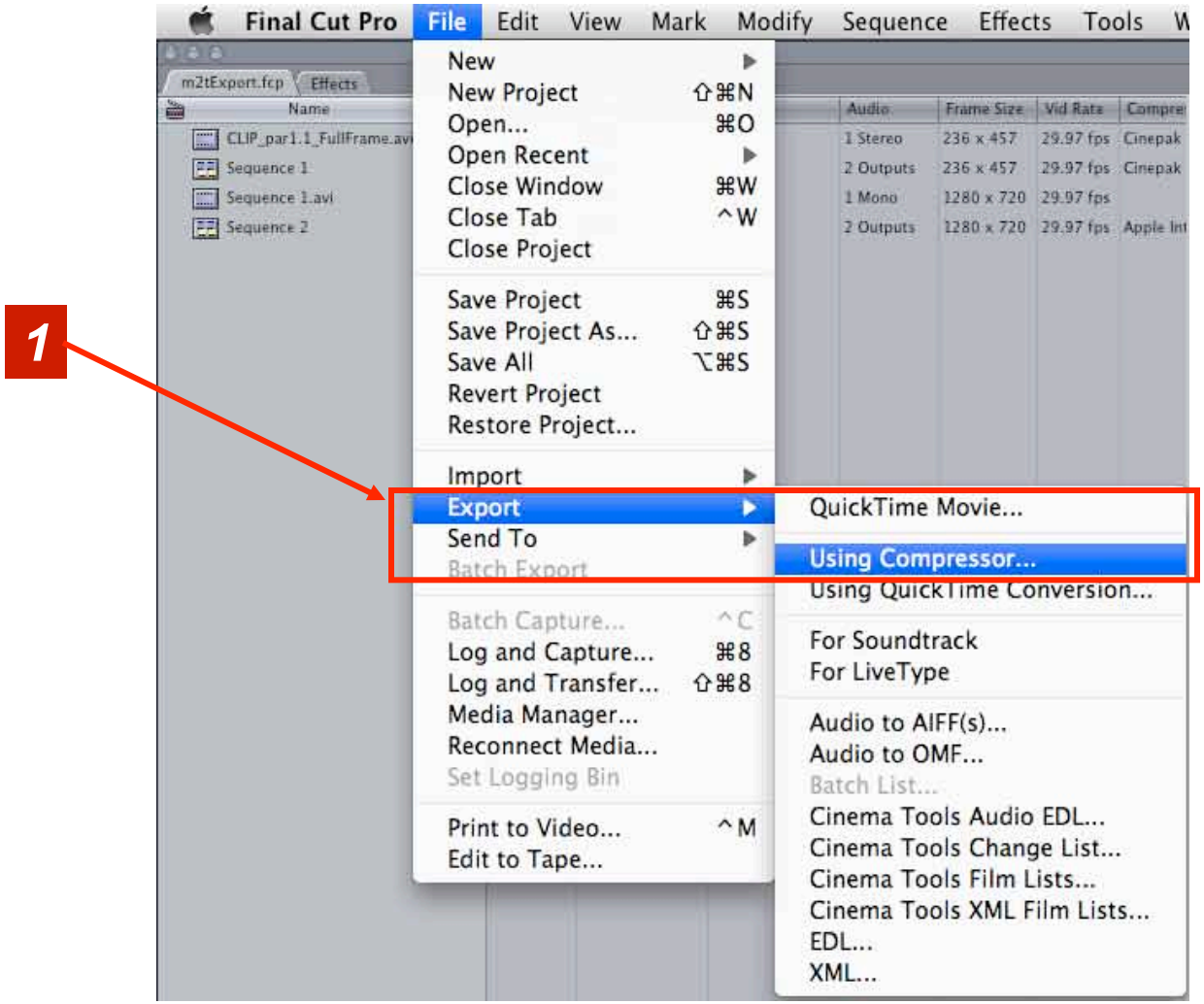


Step 8: ***The use of the VLC player to wrap the .mp4v file into the DMP recognized MPEG2-TS format will be necessary; however, transcoding is not a required procedure.***

Please read "Reformatt: MPEG4 to MPEG2 -TS" below.

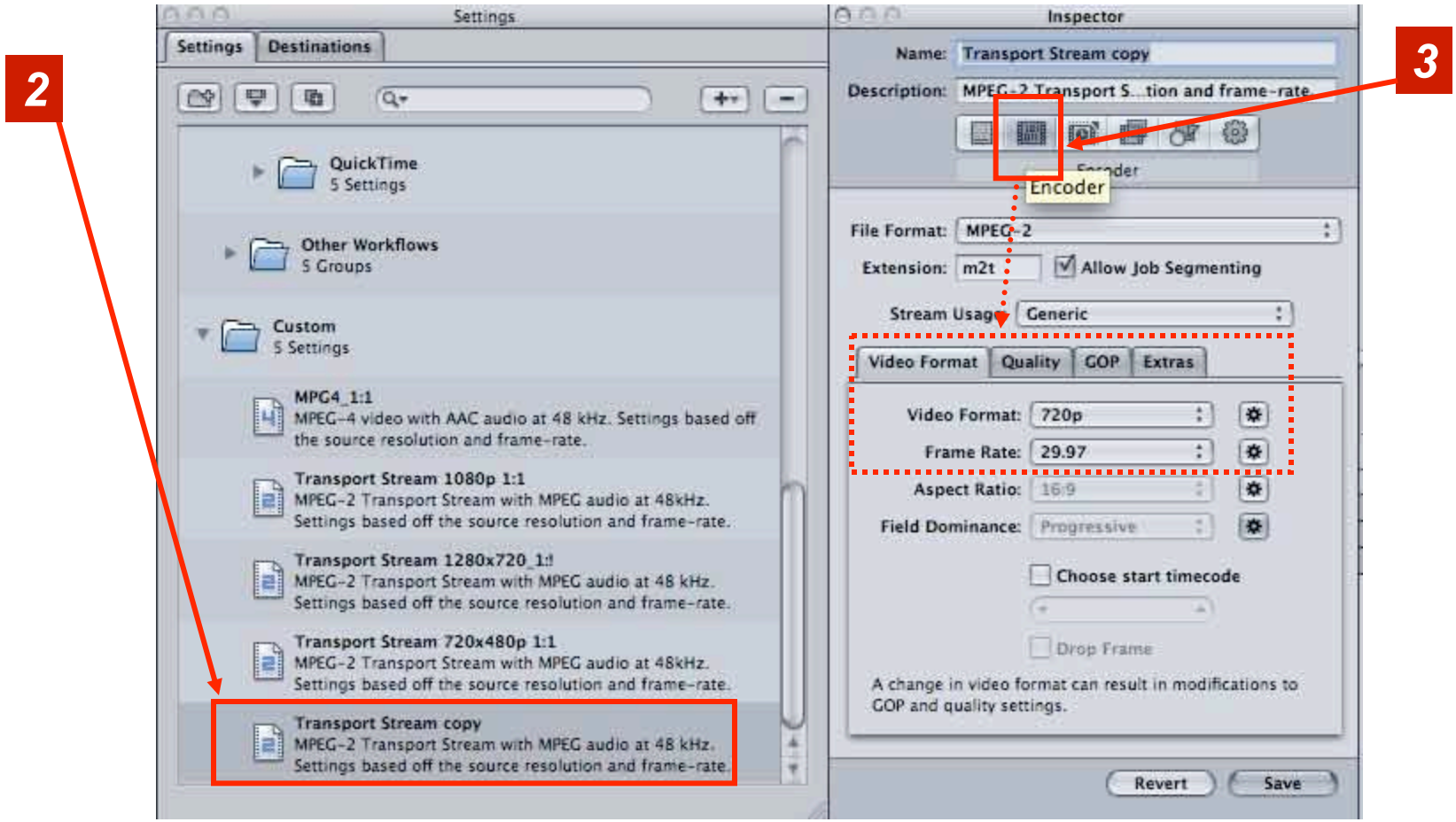
Final Cut Pro: MPEG2 Rendering

- ***Step 1: Launch*** Compressor via File>Export>Using Compressor



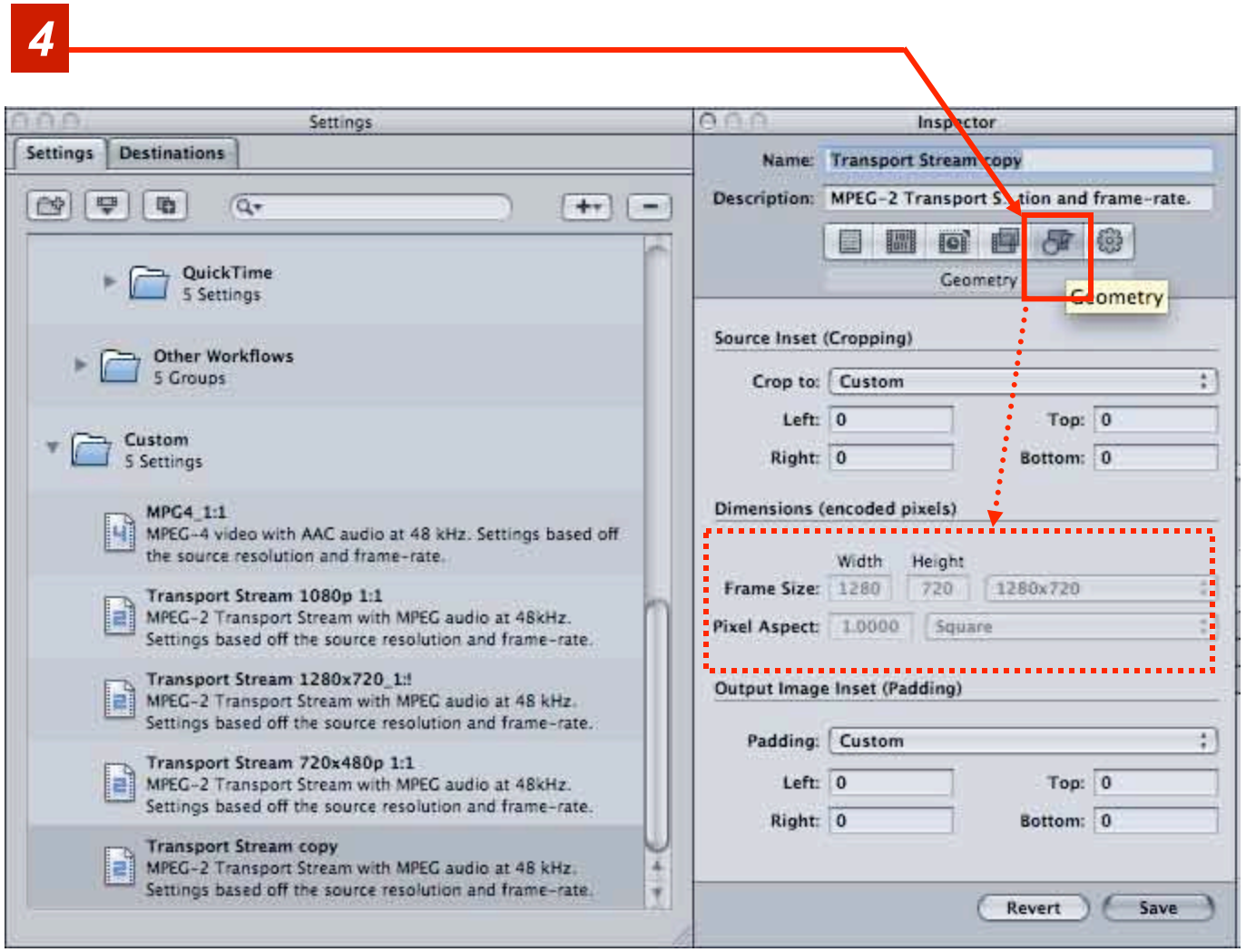
Final Cut Pro: MPEG2 Rendering

- *Step 2*: ***Duplicate*** the Mpeg-2 preset for modification.
  - *Step 3*: ***Select*** the “Encoder” button and match the settings.



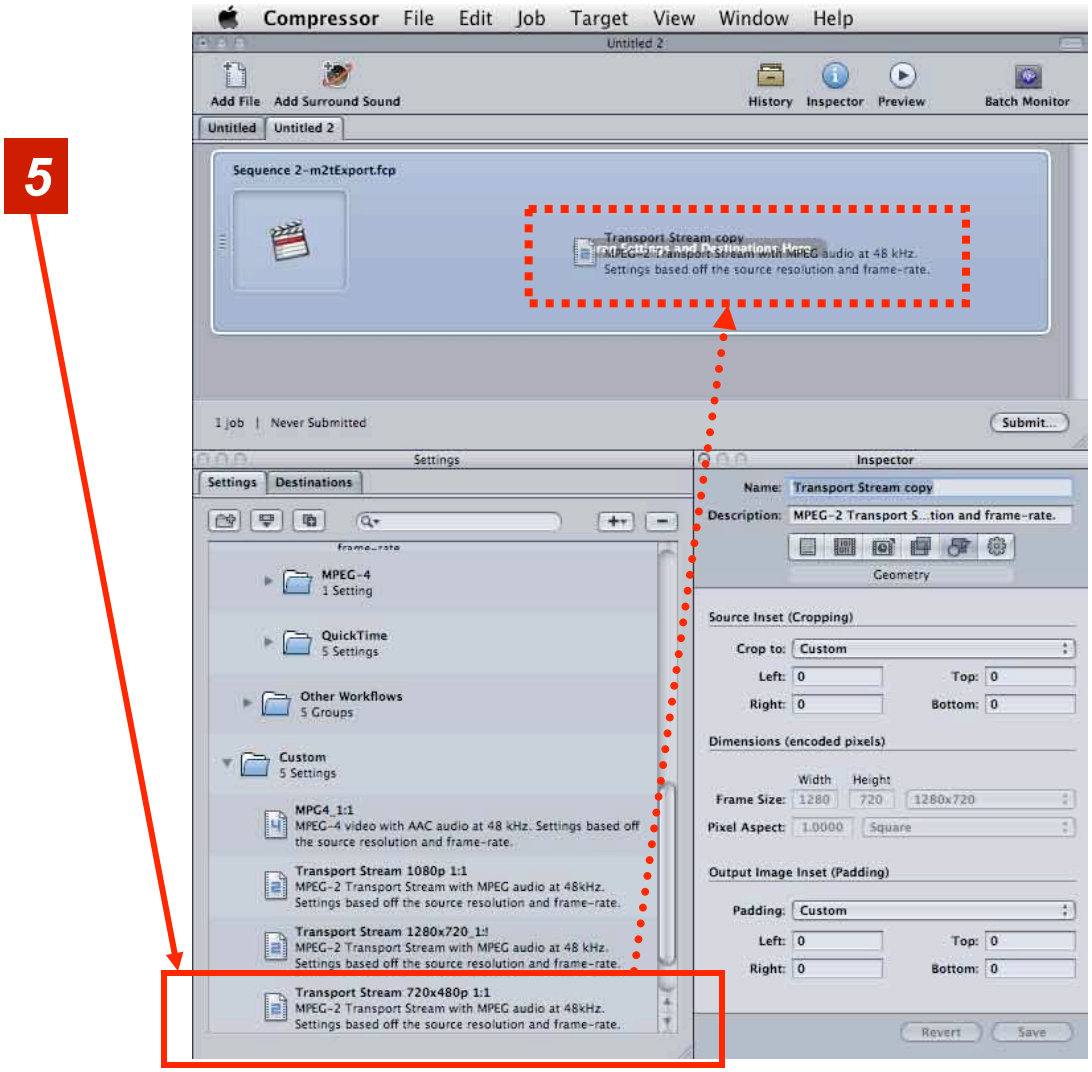
Final Cut Pro: MPEG2 Rendering

- ***Step 4*: *Select*** the “Geometry” button and verify the settings.



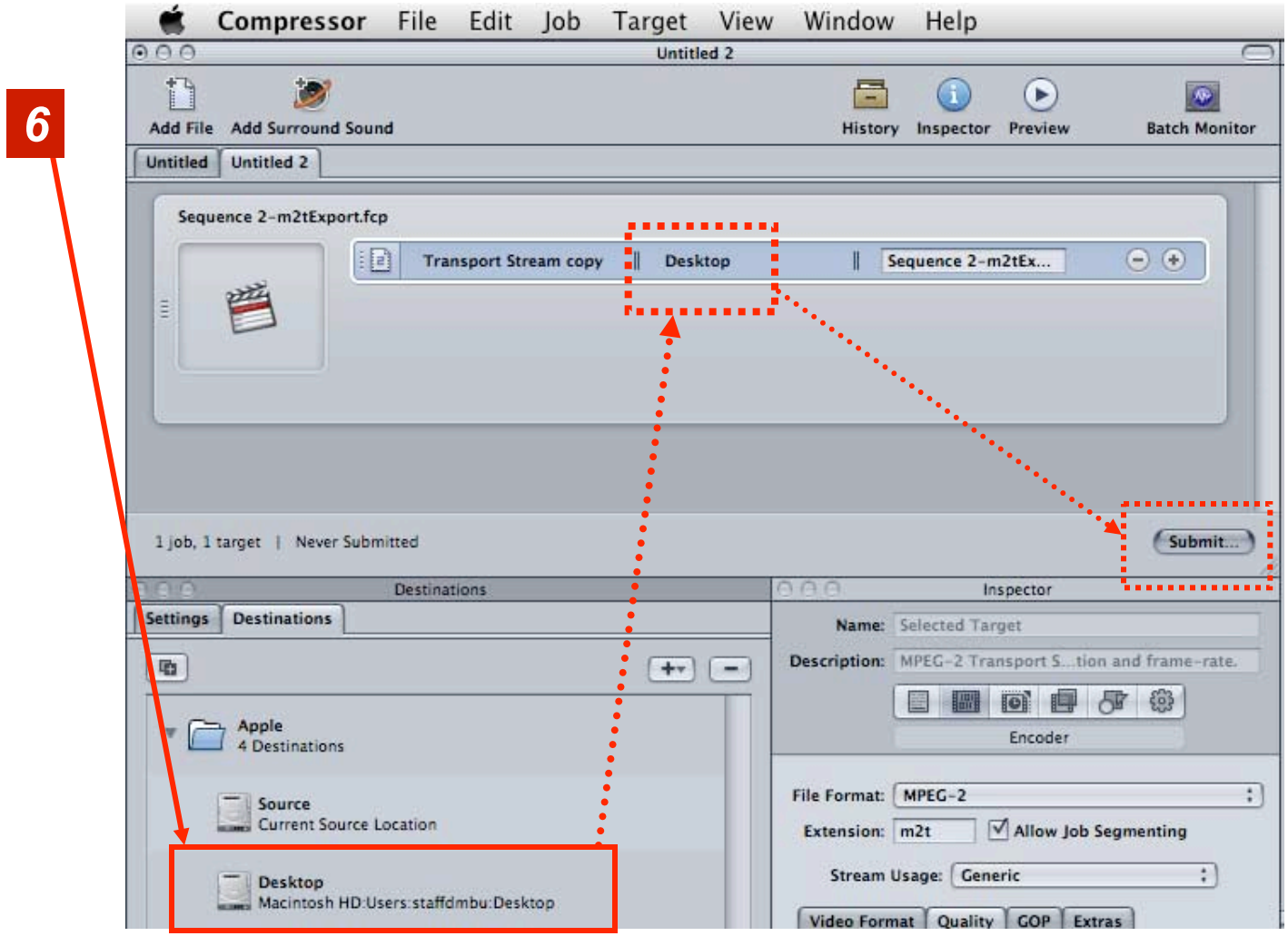
Final Cut Pro: MPEG2 Rendering

- ***Step 5*: *Drag*** the customized set to the clip.



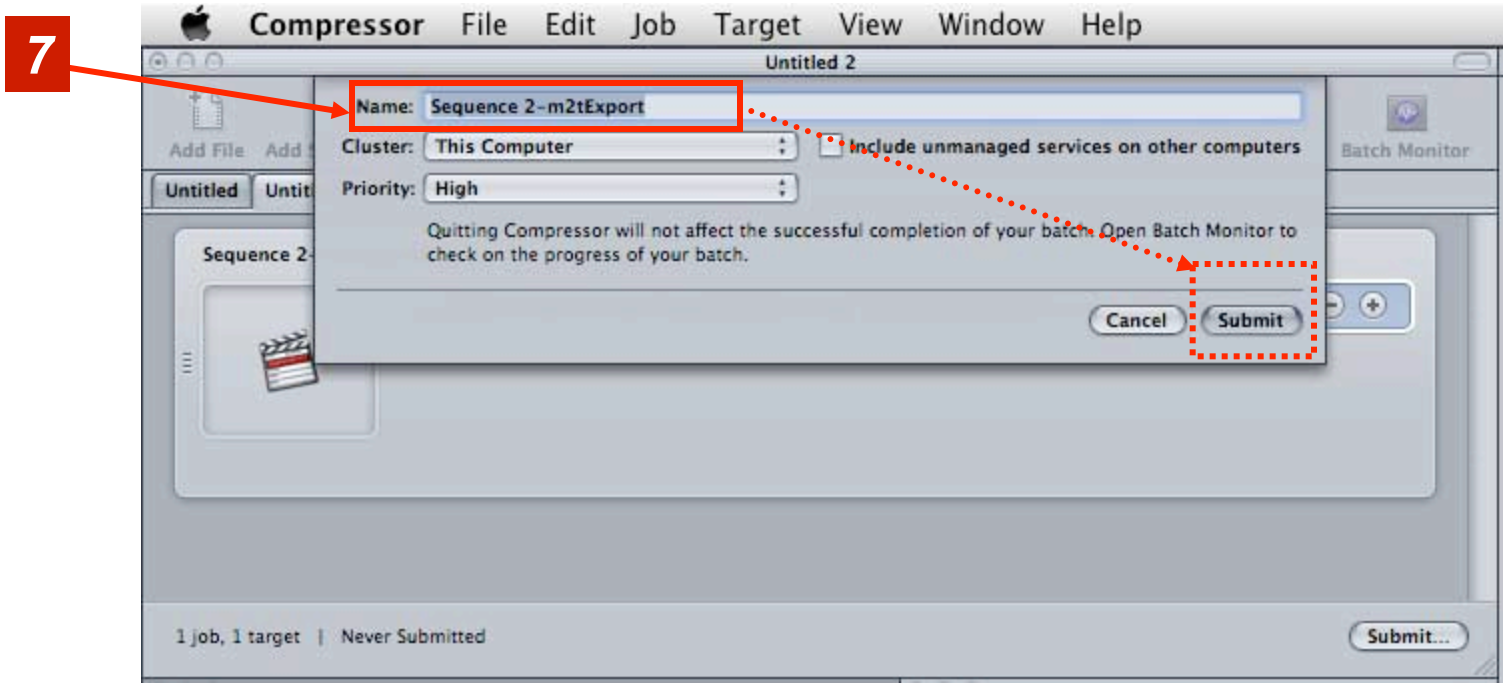
Final Cut Pro: MPEG2 Rendering

- ***Step 6*: *Drag*** a “Destination” set to the clip. Then click “Submit”.



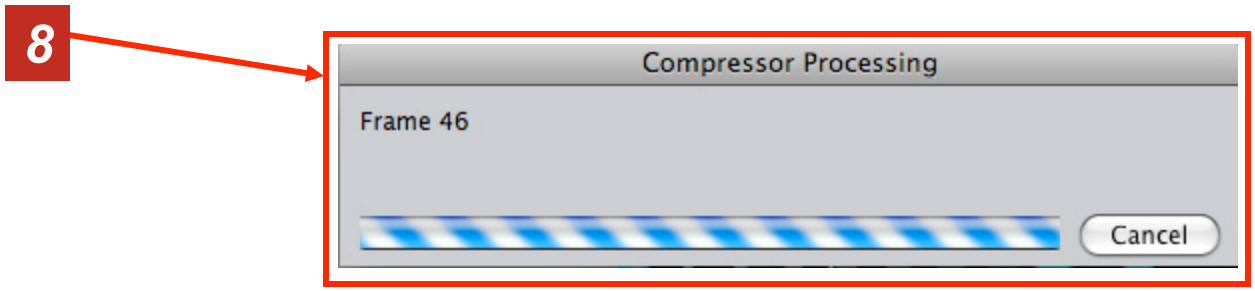
Final Cut Pro: MPEG2 Rendering

- ***Step 7*: *Enter*** a name, adding the “m2t” extension. Click “Submit”.



Final Cut Pro: MPEG2 Rendering

- *Step 8:* ***The*** dialogue box will disappear when done.

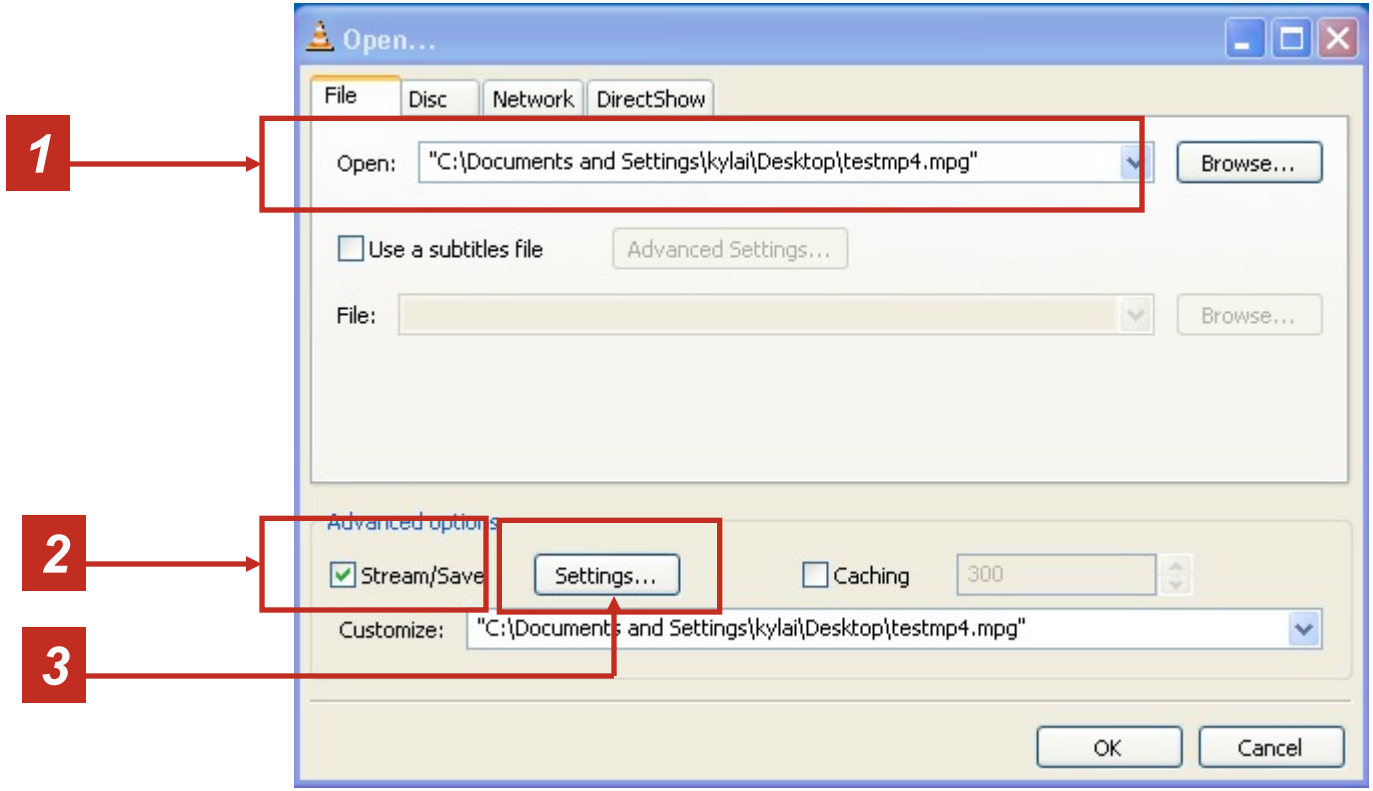


VLC: MPEG2-TS/h.264 transcoding

Step 1: Open the file you need to encode into MPEG2

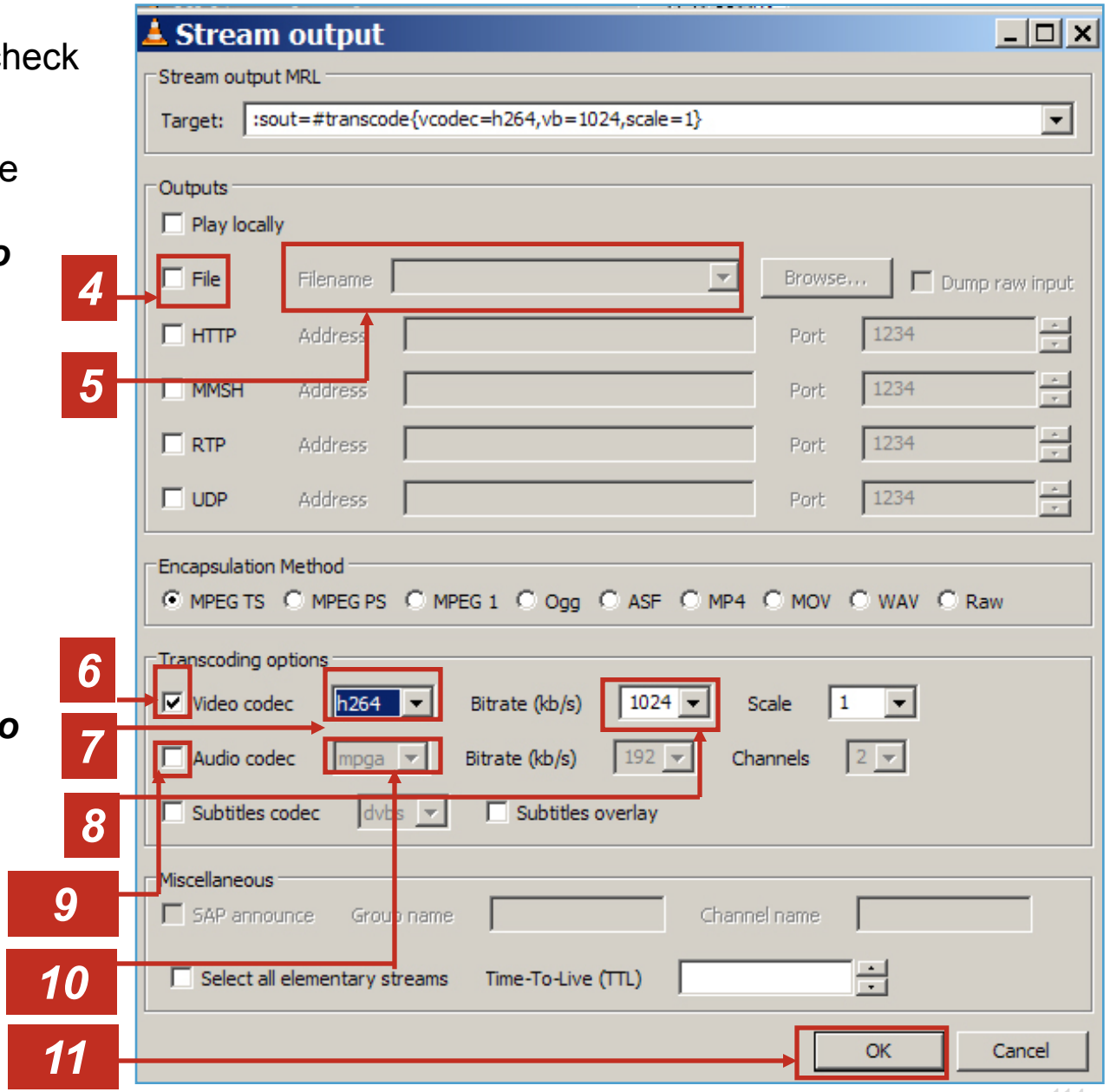
Step 2: Check "Stream/Save" check-box

Step 3: Click "Settings"



VLC: MPEG2-TS/h.264 transcoding

- ***Step 4:*** Check the ***file*** check box
- ***Step 5:*** Specify file name
- ***Step 6:*** Check the ***video codec*** check box
- ***Step 7:*** Choose ***h.264*** from the drop-down menu
- ***Step 8:*** Choose the ***bitrate*** for:
  - HD>12 -15 Mbit/sec
  - HD>5-8 Mbit/sec
- ***Step 9:*** Check the ***audio codec*** check box
- ***Step 10:*** Choose ***mpga*** from the drop -down menu
- ***Step 11:*** Hit ***OK***



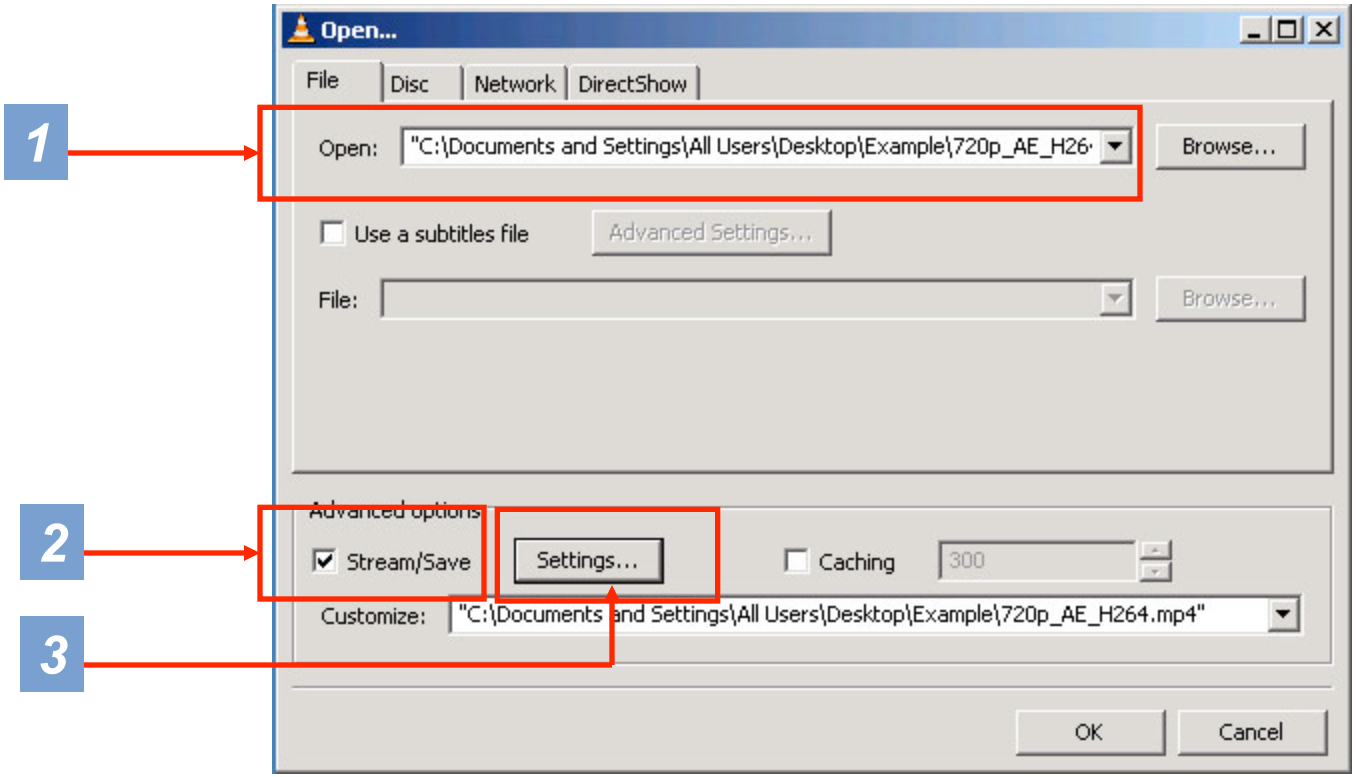
Additional Resources:

Reformatting:
MPEG4 to Mpeg2 –TS

Return to Contents

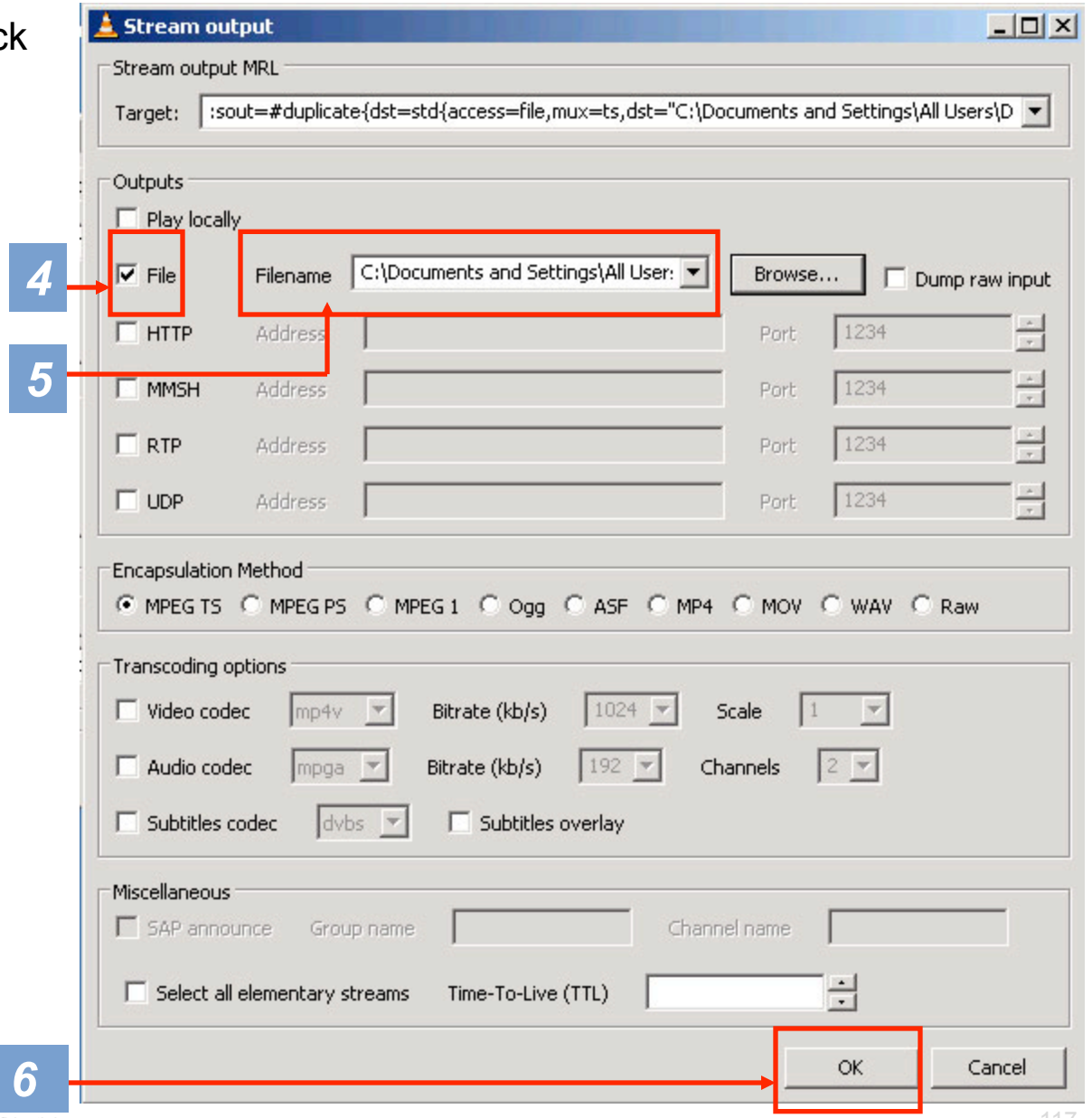
VLC: MPEG4 to Mpeg2 –TS

- ***Step 1:*** Open the file you need to re-format (wrap) into MPEG2-TS.
- ***Step 2:*** Check "Stream/Save" check-box
- ***Step 3:*** Click "Settings"



VLC: MPEG4 to Mpeg2 –TS

- ***Step 4:*** Check the ***file*** check box
- ***Step 5:*** Specify file name
- ***Step 6:*** Click “OK”.



Web and Flash Content Design Best Practices

Return to Contents

Time-Saving Tips for Stand-Alone Content Creation

- ***Step1:***

  Create a template in Photoshop:

  1. Define screen resolution.
  2. Define action and title safe area.
  3. Define the zones you are going to use for your sign project and their placement.
  4. Define placement of all elements: logo, presentation, mpeg movie, and ticker.
  5. Define color palette, type face, type size, and amount of text information.
  6. Put all design elements to show clearly the big picture (composition) of your screen design.
  7. **Please do not forget to test your design on your display.**

Time-Saving Tips for Stand-Alone Content Creation

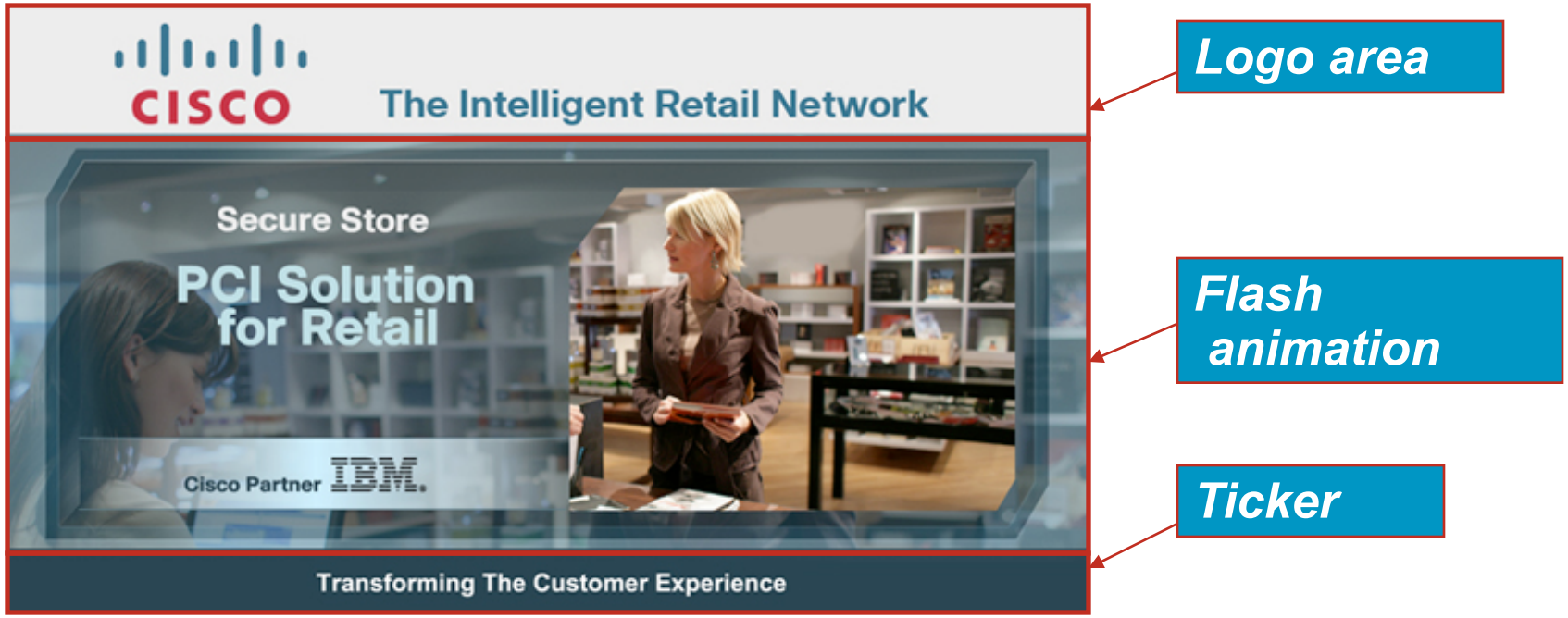
- *Step2*:

  Split the Screen identifying different media:

  a) Logo may be included into background image, or separated.
  b) Central area may contain .swf and mpg2 files, or just one of 2.
  c) Ticker line may be created in Flash, or using Java Script.

- *Step3*:

  Create html page



Screen Resolution

- Most plasma and LCD displays support ***1366x768*** resolution (in pixels). This is the size of the background image that goes into the html page.

Title and Action Safe Area

- Some of the full-screen image will be cut off on the border, and that's why it's necessary to keep approximately ***10% -20%*** of "safe area" around your design:

  10%-action safe area

  20%-title safe

Fonts

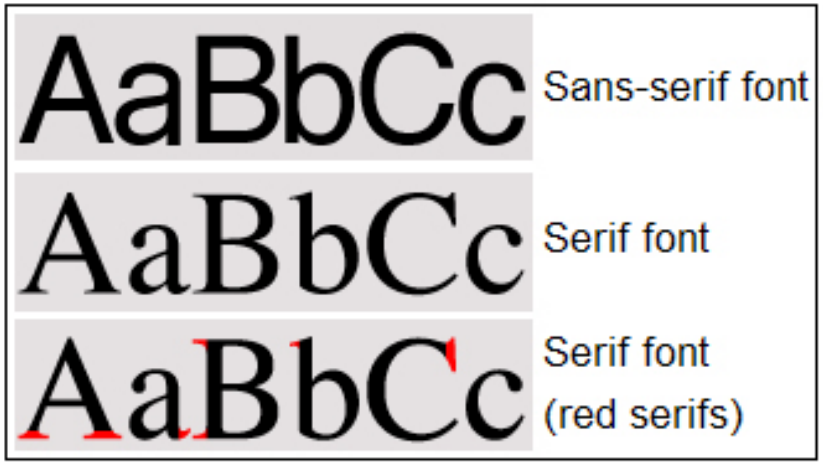
- Colors and type face for screen design:

  a) Try to avoid saturated, very bright colors.

  b) Use contrast colors/tone for text against the background color. Light text on dark background reads best.

  c) If you prefer to use a white background, make it slightly grey instead.

  d) *Italic* fonts are not recommended.



  e) Sans-serif type face fonts are recommended. Computer monitor plasmas provide a cleaner and more legible rendering of sans-serif fonts than they do for serif fonts.

Text Size & Volume
Vertical Screen Design

- Most vertically mounted plasmas will take up to approximately 26 lines of text with text size ***24 points*** for large blocks.
- Depending on the amount of text, you may reduce its ***minimum size*** to ***18 points***, however, the bigger text font, the easier to read.
- Note: If an image or text on the screen remains static or not in motion for extended lengths of time, the plasma technology may "burn in" and retain the static image on the screen – that's why it's a good idea to use image & text animations.

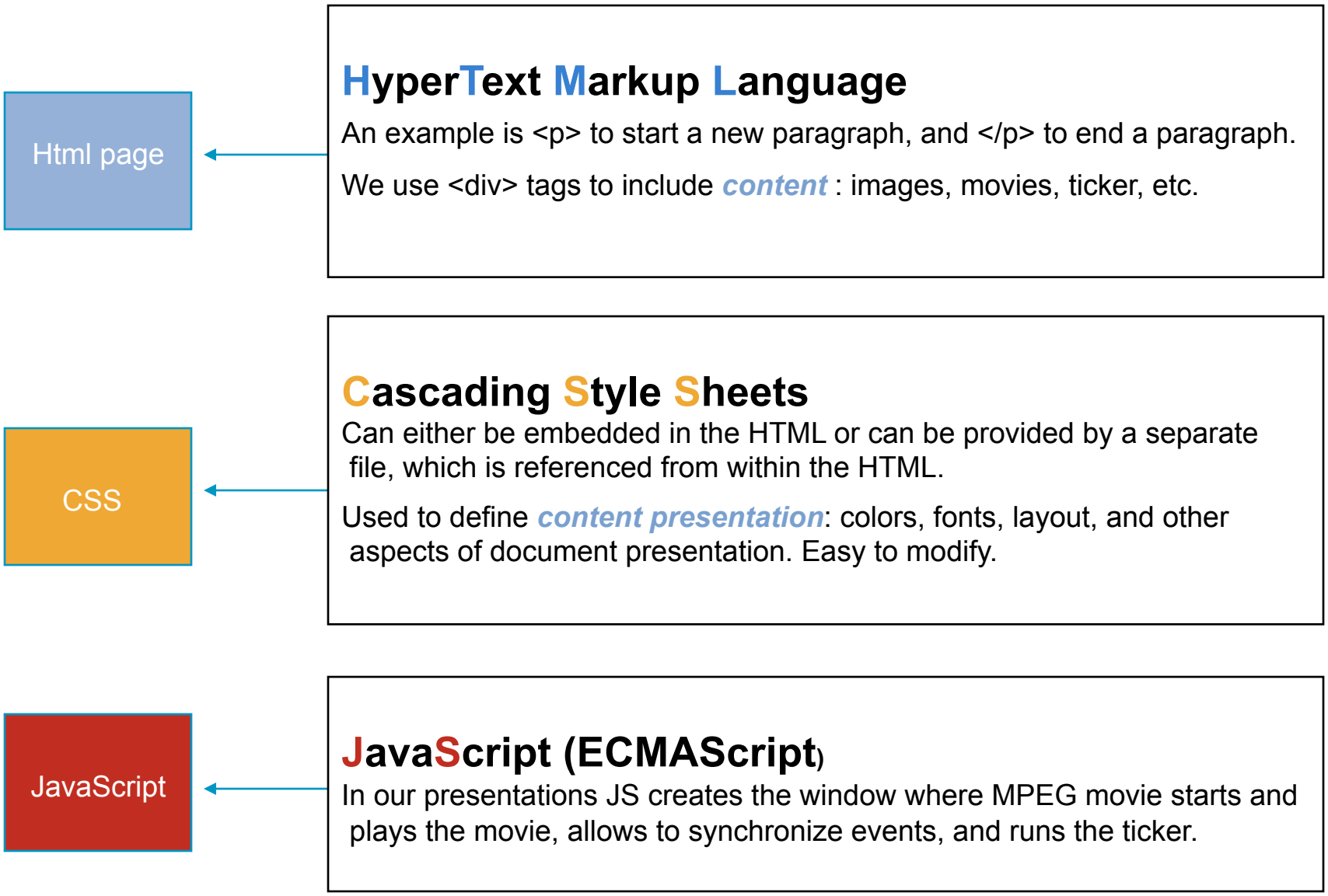
Text Size & Volume
Horizontal Screen Design

- *24 points* looks good and is easy to read on horizontally-mounted displays. With a 24 points font size, no more than 9 text lines per screen is recommended.
- Do not use too much text and to leave some negative space around the border.

How Not to Design

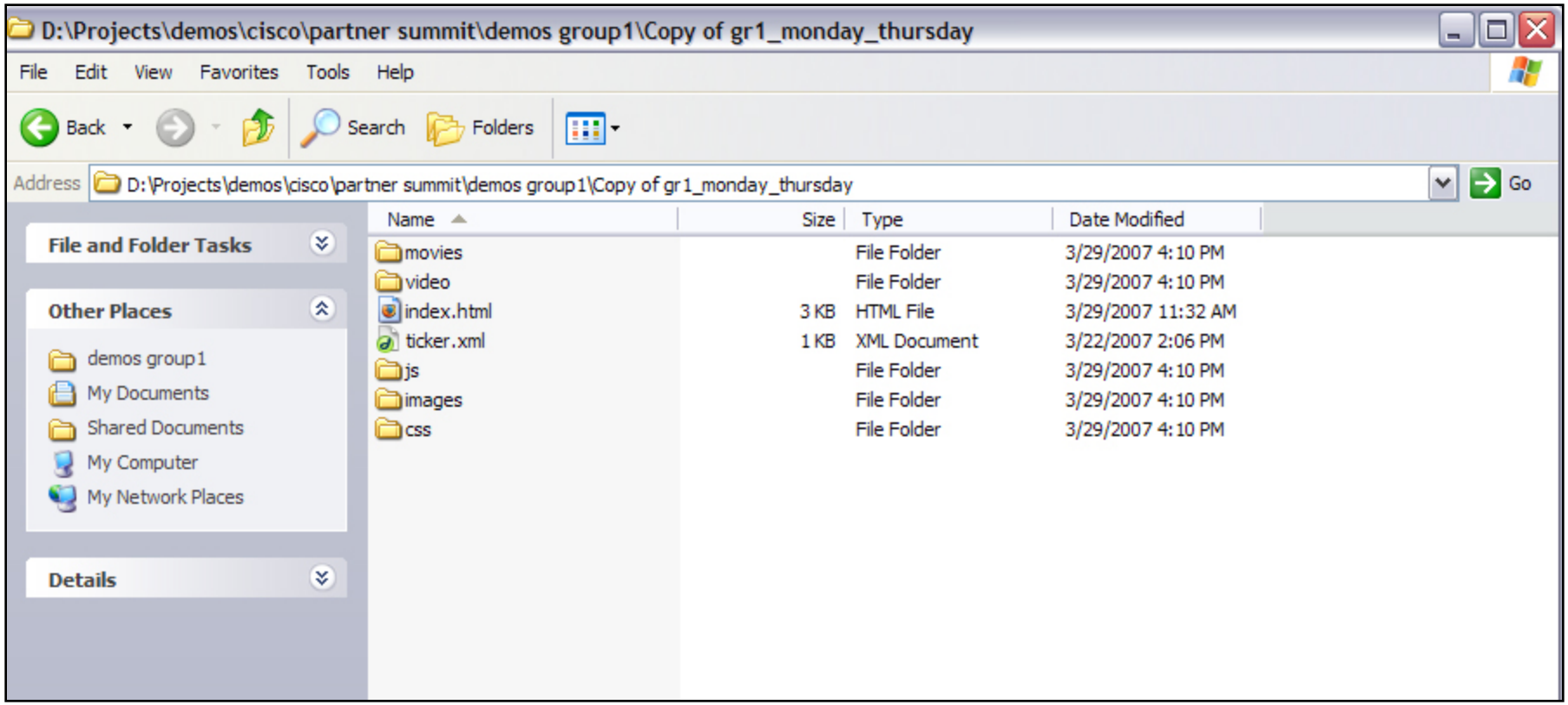
- This sign has a little too much text. You can read it if you stand close to the screen. The font size is too small at *18 points* in this design.

Design main components:



File Structures

- In this screenshot all MPEG movies are stored locally in a “movies” folder. (You can also stream them from a server).



HTML Sample Code

```
<html>
<head>
<title>Put your page name here</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
<link rel="stylesheet" type="text/css" href="css/styles.css"> <!—this line means that cascading style sheet is inked to this page- >
<script language="JavaScript" src="js/tivella.js"></script>  <!--JS that calls “init” function when the page is loaded-- >
<script language="JavaScript"  src="js/app.js" ><!—DMP Java script API- >
</script> </head>
<!—background object---- >
<body bgcolor="#000000" leftmargin="0" topmargin="0" marginwidth="0" marginheight="0"  onload="init();">
<div id= "bg" style="position:absolute; top:0px; left:0px; width:1366px; height:768px; z-index:0;"><img src="images/bg.jpg" width="1366px" height="768px" alt=""></div>
<!-- <div id="black_rect" style="position:absolute; top:197px; left:104; z-index:2;"><img src="images/black_rect.jpg"></div> -->
<!—mpeg movie object goes here --- >
<div id="movie" class="mpg_movie">
</div>
<!—Left-hand side presentation object --- >
<div id= "PP" class="PP">
<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=6,0,29,0"
      width="588px" height="492px">
  <param name="movie" value="video/animation.swf">
  <param name="quality" value="high">
  <embed src="video/animation.swf" quality="high" pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash" width="588px"
      height="492px"></embed>
</object>
</div>
<!—ticker object--- >
<div id = "bottom flash" class="bottom">
  <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" codebase="http://download.macromedia.com/pub/shockwave/cabs/flash
      /swflash.cab#version=6,0,29,0" width="1366px" height="71px">
            <param name="movie" value="video/ticker.swf">
            <param name="quality" value="high">
            <embed src="video/ticker.swf" quality="high" pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash" width="1366px"
      height="71px"></embed>
  </object>
</div>
</body>
</html>
```

Flash Objects

- There are no special tags required in the html for Flash objects.
- Html page has to contain tags for element on page: one for the ticker; one for the background image, one for mpg2 movies, one for flash movie, etc....
- Each tag has an ID. The ID is an identifier and helps in html-JS-CSS communication.
- Every element has absolute positioning on the page.
- Class "PP" is a class which has information on styles: position, width, height, etc.
- Flash animation is included into the <object> tag:

```
<object>THIS IS WHERE YOUR FLASH IS LOCATED</object>
```

You reference to your file name twice in <object> tag (marked in green below):

```
<!—flash Movie---- >
<div id= "PP" class="PP">
 <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" codebase="http:/
/download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,19,0"
 width="686px" height="987px" title="animation">
  <param name="movie" value="video/animation.swf">
  <param name="quality" value="high">
  <embed src="video/animation.swf" quality="high" pluginspage="http:/
/www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash"
 width="686px" height="987px"></embed>
 </object>
</div>
```

JavaScript – CSS – HTML Correlation

```
var mediaPlayer;            // MediaPlayer object.
var position = new Object(); // Current movie position.
var duration = new Object(); // Movie duration.
var movieTimeout = 2000;     // Movie checking interval in milliseconds.
// Put your movie URLs here.
var urlPrefix = location.href.replace( /[^\/]*$/, "" );
var playlist = new Array(
 urlPrefix + "movies/2370_digital_media_STEREO.mpg",
 urlPrefix + "movies/2370_fame_rev1cc_STEREO.mpg",
);
var movieStarted = false;
var i = 0;  // global index of a playlist
function init()     // this function is called when the page is loaded
{
 mediaPlayer = new tvMediaPlayer();
 mediaPlayer.osdSetAlpha(0, 0, 1366, 768 , 255); // you may change the numbers 1366(width) and 768(height) to your screen resolution

 movie();
}
function movie()   // this function tells about all interactions that happen with MPEG movies on page.
{
 var el;
 el = document.getElementById("movie"); // this sets an ID to element (MPEG movie)
 if (el)
   mediaPlayer.setOutputElement(window, el, 0);
  if (!movieStarted) {
    mediaPlayer.stop();
         movieUrl = playlist[i];
    mediaPlayer.play(movieUrl);
    movieStarted = true;
    ++i;
         if (i >= playlist.length)
      i = 0;
 } else {
      mediaPlayer.getPosition(position);
      mediaPlayer.getDuration(duration);
      if (position.value < 0 || duration.value <= 0 || (duration.value == position.value && position.value != 0))
        movieStarted = false;
  }
  window.setTimeout("movie();", movieTimeout);
}
```

CSS code

```
.mpg_movie {
position:absolute;
top:238px;
left:729px;
width:521px;
height:390px;
z-index:20;
}
```

HTML page code

```
<!-mpeg movie object goes here ->
<div id="movie" class="mpg_movie">
</div>
```

Note: You can copy and paste this example into your text editor.

CSS Sample

```
/* CSS Document */

.no_border {
  margin: 0px;
}
.logo
{position:absolute;
top:0px;left:0px;}

.mpg_movie {
position:absolute;
top:238px;
left:729px;
width:521px;
height:390px;
z-index:20;
}

.PP
{
position: absolute;
top:178px;
left:52px;
z-index:2;}
.bottom
{
position:absolute;
top:701px;
left:0px;
z-index:30;
}
```

Note: You can copy and paste this example into your text editor.

On-Screen Synchronization

- JavaScript (JS) can be used to create a MPEG zone on a html page; and to control its properties and actions—synchronize different events, rotate images and movies, etc.
- MPEG and images are synchronized using JS in this demo. The JS triggers a new image to appear on the right whenever a new MPEG video plays on the left.

MPG2 movie

image

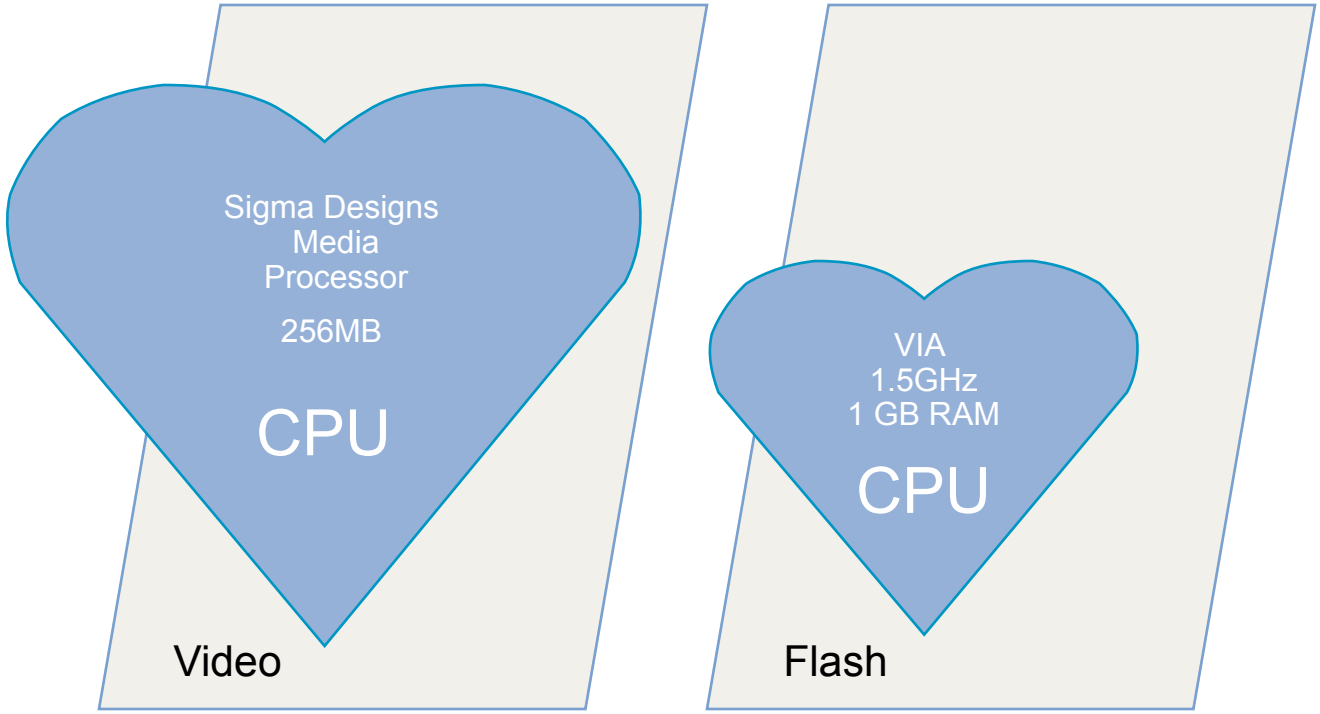
Javascript Code Created for Synchronization

```
var slides = new Array(                          //we have 4 slides in slides Array that need to be synchronized with 4 MPEG movies Array
  new Slide("inc/slides/01_01.png", 402000),  //the number defines for how long each slide plays (in milliseconds)
  new Slide("inc/slides/01_02.png", 1202000), //each slide is specified MPEG movie duration
  new Slide("inc/slides/01_03.png", 402000),
  new Slide("inc/slides/01_04.png", 400000)
);

   // Slides related stuff.
  if (slides.length)
    _slide = document.getElementById("slide"); // slide is given an ID to which html refers

function slide( idx )
{
  if ( idx != null )
    _slide.src = slides[ idx ].url;
  else {
                        if (slides.length > 1) {
                                        _slide.src = slides[_slideI].url;

                                        if (slides[_slideI + 1].url != undefined)
                                        window.setTimeout("slide();", slides[_slideI].time);
                        else {
                                        ++_slideI;
                                        if (!onDevice())
                                        window.setTimeout("slide();", _movieTimeout);
                        }

                                        if (++_slideI >= slides.length)
                                                        _slideI = 0;
                        }
  }
}
```

Note: You can copy and paste this example into your text editor.

Flash Content Creation

- For flash content creation, the designer should realize that any hardware, such as the Digital Media Player, has its processing limitations. The DMP has ***2 CPUs***:



Flash Content Creation
General Guidelines

- Flash audio is not supported in these versions.
- Animations of small objects & small movie clips with little movement work very well. Many small animations will work better than one big file if there are different tyoes of movements. Max recommended size 640pxX 480px.
- Please do not create more than one effect in the same segment of your movie time line.
  - For example, if you already have a fading in and out effect with an object, please do not use resizing at the same time
  - Or if you want to create effects for several objects, please make sure these effects do not happen simultaneously
- Please do not use large resizing at any time.
- Try not to use shape tweening at all–or if you must, use it only on very small shapes. Tweening slows down the processor and animations will not play well on DMP.
- The Flash player works better when you split the screen into different files, like a ticker and a presentation, versus keeping the files in one flash file.
- Current **Flash Player9, Player6** and **Flash Player7** do not support .flv file format. Flash Video files have to be converted into MPEG2.

Flash Content Creation
General Guidelines

- Please use 12 fps frame rate.
- Imported video plays back very slowly. It's better not to use it at all, or if you have to, make the video window small and make sure that nothing else happens in animation at the same time.
- If you use bitmaps for your movie, make sure they are not huge in size: like print quality resolution, and not small, loose quality images.
- Depending on the required movie output size, animated images that are embedded in the movie have jpeg compression and should not exceed 600KB in file size.
- Images that are not embedded in the Flash (but referenced by the Flash---referred to as "external data") should not exceed 5 MB in file size. Image resolution 72 dpi.
- Current DMP Browser (Mozilla Firefox) does not support transparent swf files.
- Alpha transparency works better on small objects.

Flash Content Creation

- If you want to use a Flash movie with external text data and need particular fonts to be displayed, please embed this font into your movie.
- How to embed fonts: You can embed any font to static text by following the instructions below. This is better method of embedding text than converting the text into a shape in order to keep chosen type face.
- External jpegs in .swf movies have to be non-progressive scan.

Font/Character embedding into flash movie:

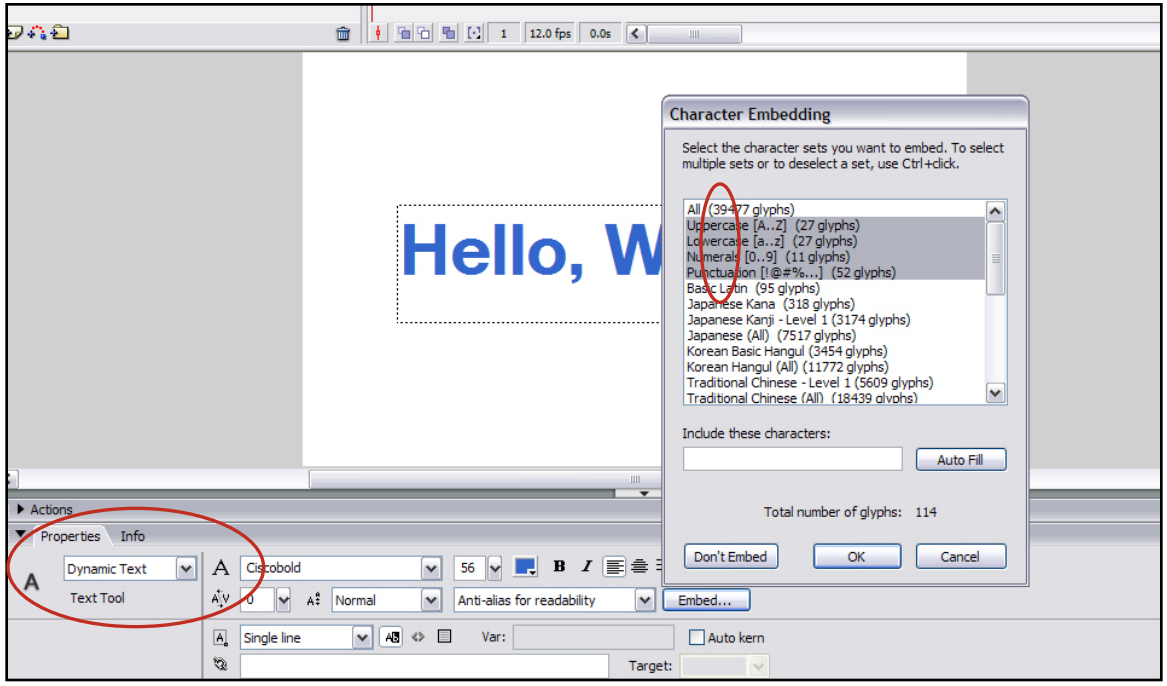
1: create your text

2: apply Dynamic Text property to it

3: click the Embed… button

4: select characters that need to be embedded

5: hit the OK button

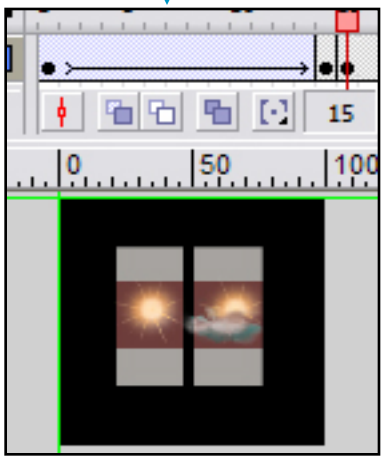


Flash Content Creation: Tips and Tricks for Alpha Transparency

- If you use Alpha transparency, it's best to not use the whole range from 0% to 100%. The CPU in this case has to work too hard. Try to shorten this range. You can alter Alpha transparency with brightness for fading out effect.
- Fade-in and fade-out effect can be used in 3 key-frames. Additional, 3rd key frame helps to improve DMP performance.

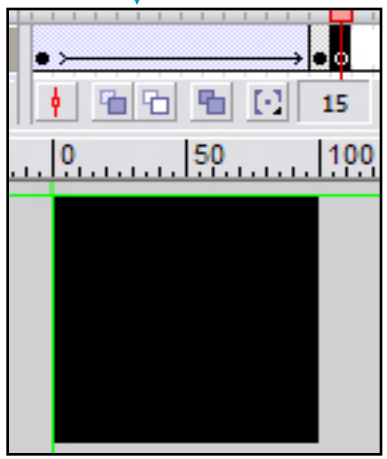
Fade-in:

1) 1st key frame: alpha is set to 80%
2) 2nd key frame: alpha is set to 90%
3) 3rd key frame: color is set to "none"



Fade-out:

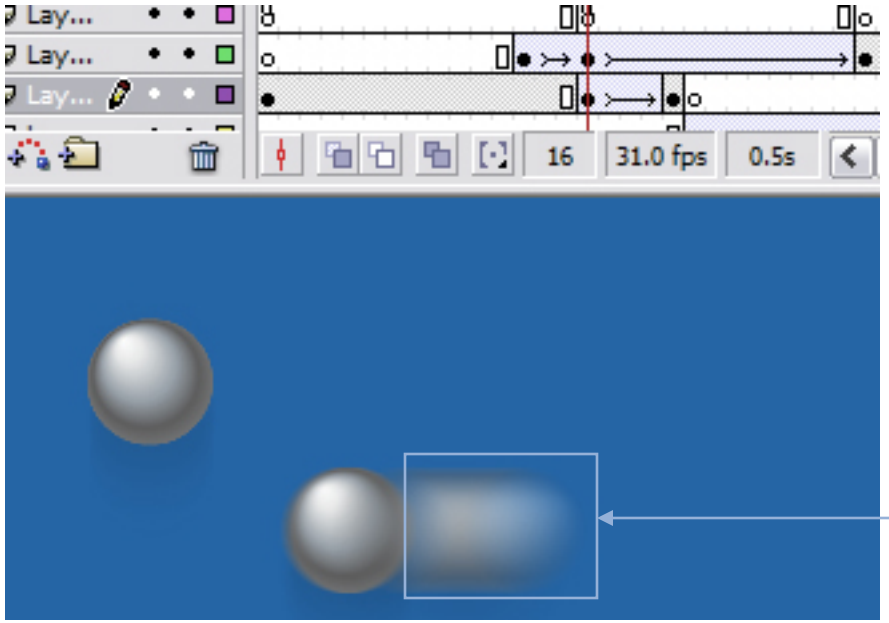
1) 1st key frame: alpha is set to 90%
2) 2nd key frame: alpha is set to 80%
3) 3rd key frame: empty key frame



Note: suggested 80% and 90% numbers are optional, and may vary depending on the image size. Experiment with your device to get best results

Flash Content Creation: Tips and Tricks Reducing Jitter

- If you want to move an object in your animation, and your object's movement is "jittery", try to reduce the amount of frames in between starting and ending key frames of motion tweening and use an additional jpeg image with motion blur filter applied to it in Photoshop. Use this blurred image to create an effect of fast movement.



Motion Blur filter was applied to jpeg in Photoshop, imported into flash and used to "fake" fast movement

Note: On smaller objects you'll get better performance

Flexible Web Design

Return to Contents

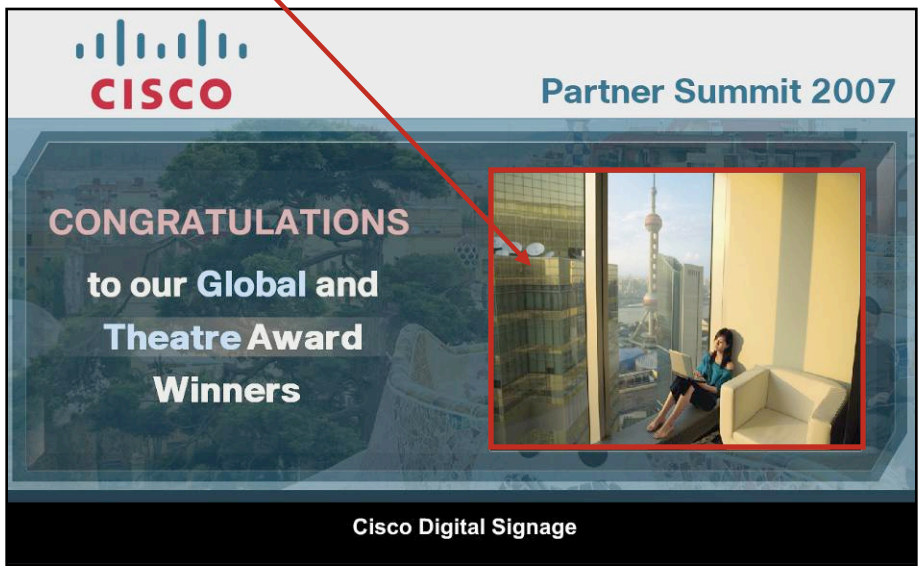
What is Modular Web Design?

- Flash or web pages that contain variables which refer to external text or XML files, where constantly-changing data can be easily changed.

You can rotate as many images as you want, just add them to your images folder and Javascript

Text is created in Flash with an external xml file— can be easily replaced or modified

Javascript is used to rotate images in this design

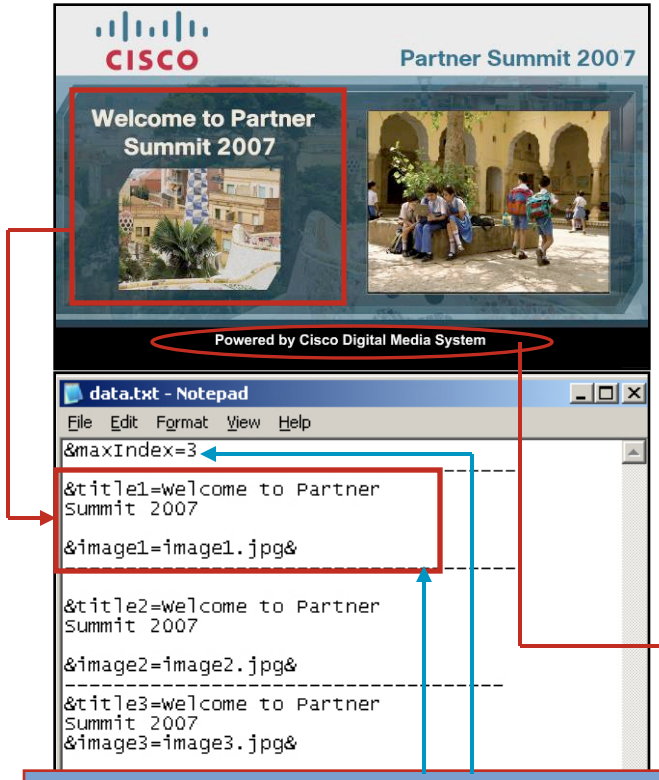


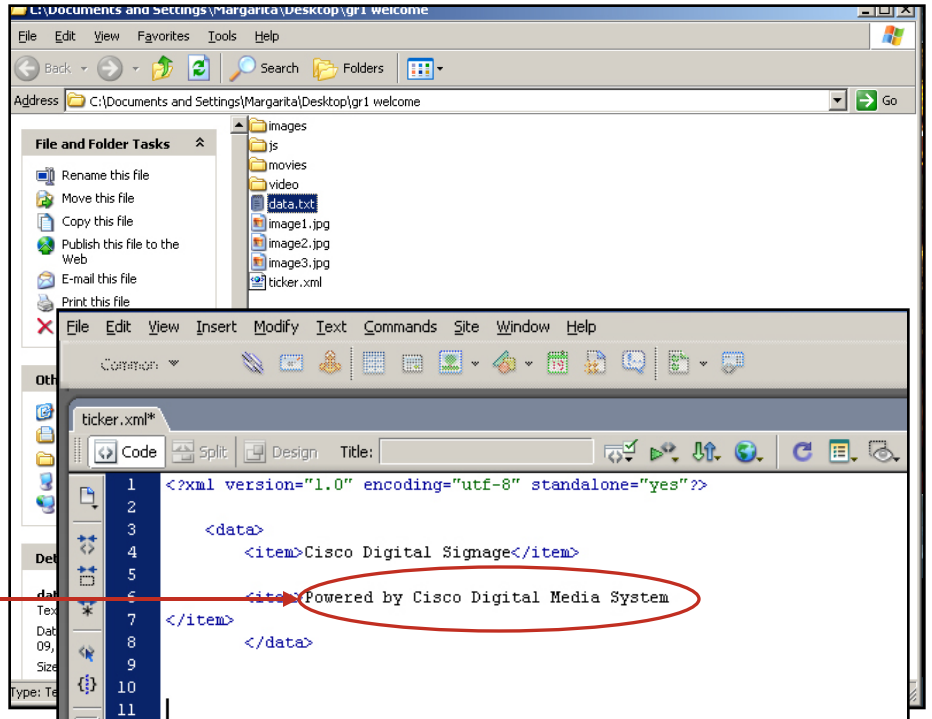
Why Use Modular Web Design?

- Pre-designed modular template allows for any user to easily make changes to signage data and content by modifying text or XML files
- Allows for control of brand consistency across web pages regardless of page content
- Less maintenance on core web page and Flash design

Sample Modular Web Design

- In this presentation, it's also easy to make updates by just replacing external data—data.txt contains editable text and information used in flash movies, and ticker.xml contains editable text data for the ticker.





There are 2 different elements for each left-hand side page: the title and the image.
title1 – title that you want on the 1st page, etc.
Image1 – corresponding image displayed with the title.
maxIndex = 3 means that there are 3 pages that will rotate through the animation.

This is an editable .xml file. Just replace the text in the red circle with your own text.

Designing for Varying Screen Sizes

- This design template fits on any screen size because the frame and window sizes are not fixed. They are specified as a percentage of the screen or page area.

Html, Javascript, CSS code will be shown on following after this pages

HTML for Varying Screen Sizes

- Sample HTML for non-fixed page design

```
<html>
<head>
<title>..:: Pixar ::..</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
<link rel="stylesheet" type="text/css" href="inc/color.css">

<script language="JavaScript" src="js/items.js"></script>

<script language="JavaScript" src="js/movies.js"></script>
<script language="JavaScript" src="js/slides.js"></script>

<script language="JavaScript" src="js/tivella.js"></script>
<script language="JavaScript" src="js/app.js"></script>
<link rel="stylesheet" type="text/css" href="css/styles.css">

</head>

<body onLoad="init();" bgcolor="blue">
<img id="bg" class="bg" src="images/bg.jpg" alt="">

<div id="movie" class="movie"></div>

<div id="slide" class="slide" alt="" src="">
  <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" codebase="http://download.macromedia.com/pub/shockwave/cabs/flash
/swflash.cab#version=7,0,19,0" width="100%" height="100%" title="fla">
    <param name="movie" value="video/flash_ads.swf">
    <param name="quality" value="high">
    <embed src="video/flash_ads.swf" quality="high" pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash"
 width="100%" height="100%"></embed>
  </object>
</div>

<div id = "bottom_flash" class="bottom">
  <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000" codebase="http://download.macromedia.com/pub/shockwave/cabs/flash
/swflash.cab#version=6,0,29,0" width="100%" height="100%">
              <param name="movie" value="video/ticker.swf">
              <param name="quality" value="high">
              <embed src="video/ticker.swf" quality="high" pluginspage="http://www.macromedia.com/go/getflashplayer" type="application/x-shockwave-flash"
 width="100%" height="100%"></embed>
  </object>
</div>

</body>
</html>
```

Note: You can copy and paste this example into your text editor.

Javascript for Varying Screen Sizes

- Sample Javascript for non-fixed page design

```
var _mediaPlayer;          // MediaPlayer object.
var _position = new Object(); // Current movie position.
var _duration = new Object(); // Movie duration.
var _movieTimeout = 1234;     // Movie checking interval in milliseconds.
var _movieStarted = false;
var _slide;
var _movie = 0;
var _slidel = 0;

function onDevice()
{
  return navigator.platform != "Win32";
}

function adjust()
{
  var el;

  el = document.getElementById("bg");
  if ( el ) {
    el.style.setProperty("width", window.innerWidth + "px", "important");
    el.style.setProperty("height", window.innerHeight + "px", "important");
  }

  var kx = window.innerWidth / 1366;
  var ky = window.innerHeight / 768;

  el = document.getElementById("movie");
  el.style.setProperty("left", 728 * kx + "px", "important");
  //el.style.setProperty("top", 235 * ky + "px", "important");
  el.style.setProperty("top", 235 * ky + "px", "important");
  el.style.setProperty("width", 520 * kx + "px", "important");
  //el.style.setProperty("height", 392 * ky + "px", "important");
  el.style.setProperty("height", 386 * ky + "px", "important");

  el = document.getElementById("slide");
  el.style.setProperty("left", 119 * kx + "px", "important");
  el.style.setProperty("top", 288 * ky + "px", "important");
  el.style.setProperty("width", 521 * kx + "px", "important");
  //el.style.setProperty("height", 392 * ky + "px", "important");
  el.style.setProperty("height", 280 * ky + "px", "important");

  el = document.getElementById("bottom_flash");
  el.style.setProperty("left", 0 * kx + "px", "important");
  el.style.setProperty("top", 701 * ky + "px", "important");
  el.style.setProperty("width", 1366 * kx + "px", "important");
  //el.style.setProperty("height", 392 * ky + "px", "important");
  el.style.setProperty("height", 84 * ky + "px", "important");

 /* el = document.getElementById("ticker");
  el.style.setProperty("left", "0px", "important");
  el.style.setProperty("top", 683 * ky + "px", "important");
  el.style.setProperty("width", 1366 * kx + "px", "important");
  el.style.setProperty("height", 84 * ky + "px", "important");*/

//  el = document.getElementById("movie");
//  el.style.setProperty("left", "0px", "important");
//  el.style.setProperty("top", "0px", "important");
//  el.style.setProperty("width", "1000px", "important");
//  el.style.setProperty("height", "700px", "important");
}
function init()
{ var el;
```

app.js

```
// Put your movie URL here.
var urlPrefix = location.href.replace( /[^\/]*$/, "" );
var movies = new Array(
urlPrefix + "video/oil_sands_mp2a.mpg",
urlPrefix + "video/oil_sands_mp2a.mpg"


);
```

movies.js

Specify the width and the height of your screen here. (1336 is the width and 768 is the height of the screen)

The “adjust” function in this code is adjusting all elements on the page accordingly to your screen resolution. You can copy and paste this code into your text editor.

Note: You can copy and paste this example into your text editor.

CSS for Varying Screen Sizes

- Sample CSS for non-fixed page design

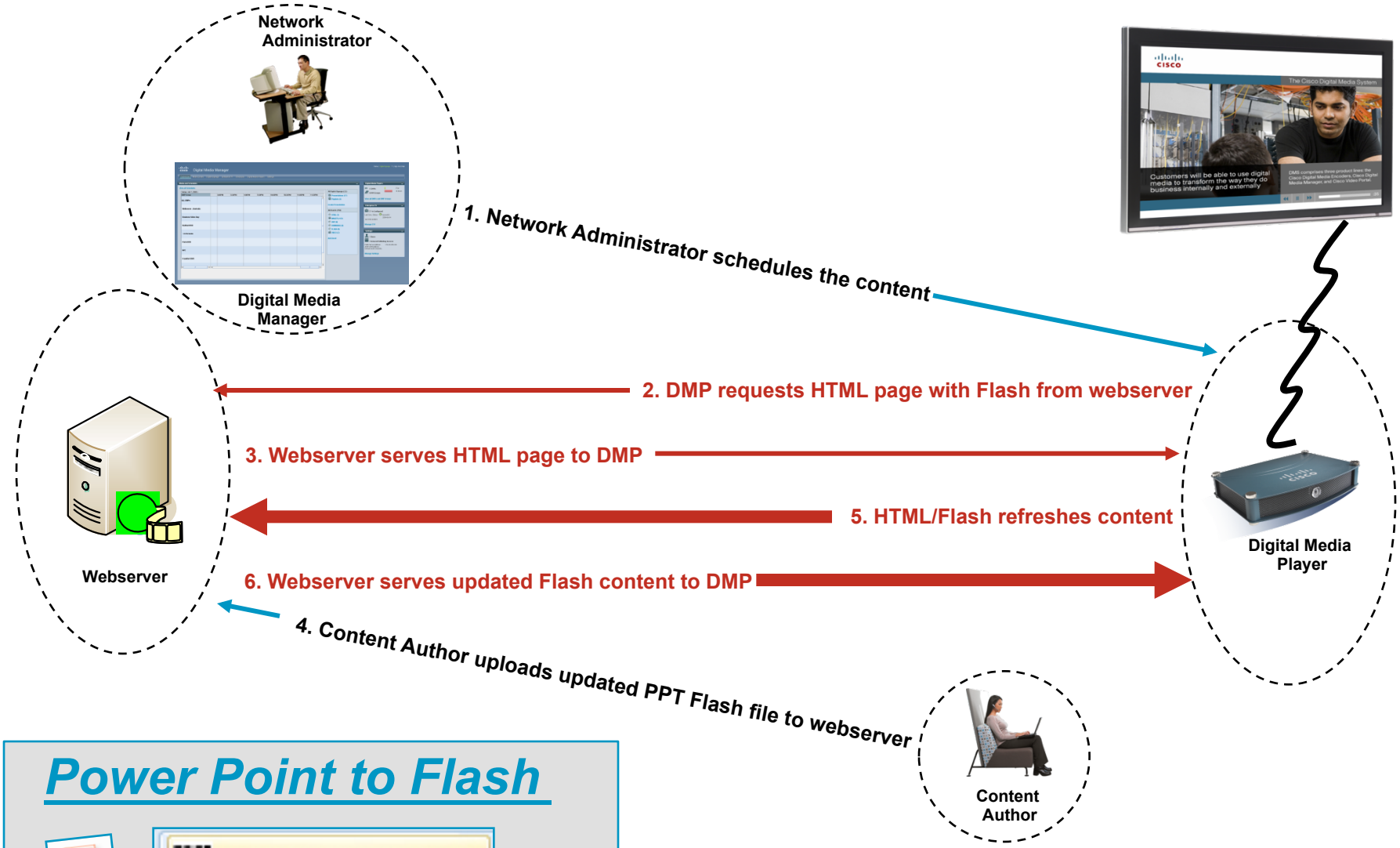
```
/* CSS Document */
body {
  background-color: #000000;
  margin: 0px;
}

.bg {
  position: absolute;
  top: 0px;
  left: 0px;
  width: 100%;
  height: 100%;
  z-index:0;
}

.movie {
  position: absolute;
  top: 235px;
  left: 728px;
  width: 100%;
  height: 100%;
 }

.slide {
  position: absolute;
  top: 235px;
  left: 119px;
  width: 100%;
  height: 100%;
  z-index: 2;
}

.ticker {
  position: absolute;
  top: 683px;
  left: 0px;
  z-index: 30;
  width: 100%;
  height: 100%;
  font-weight: bold;
  font-family: Arial, Helvetica, sans-serif;
  text-align: center;
  vertical-align: middle;
}
```

Note: You can copy and paste this example into your text editor.

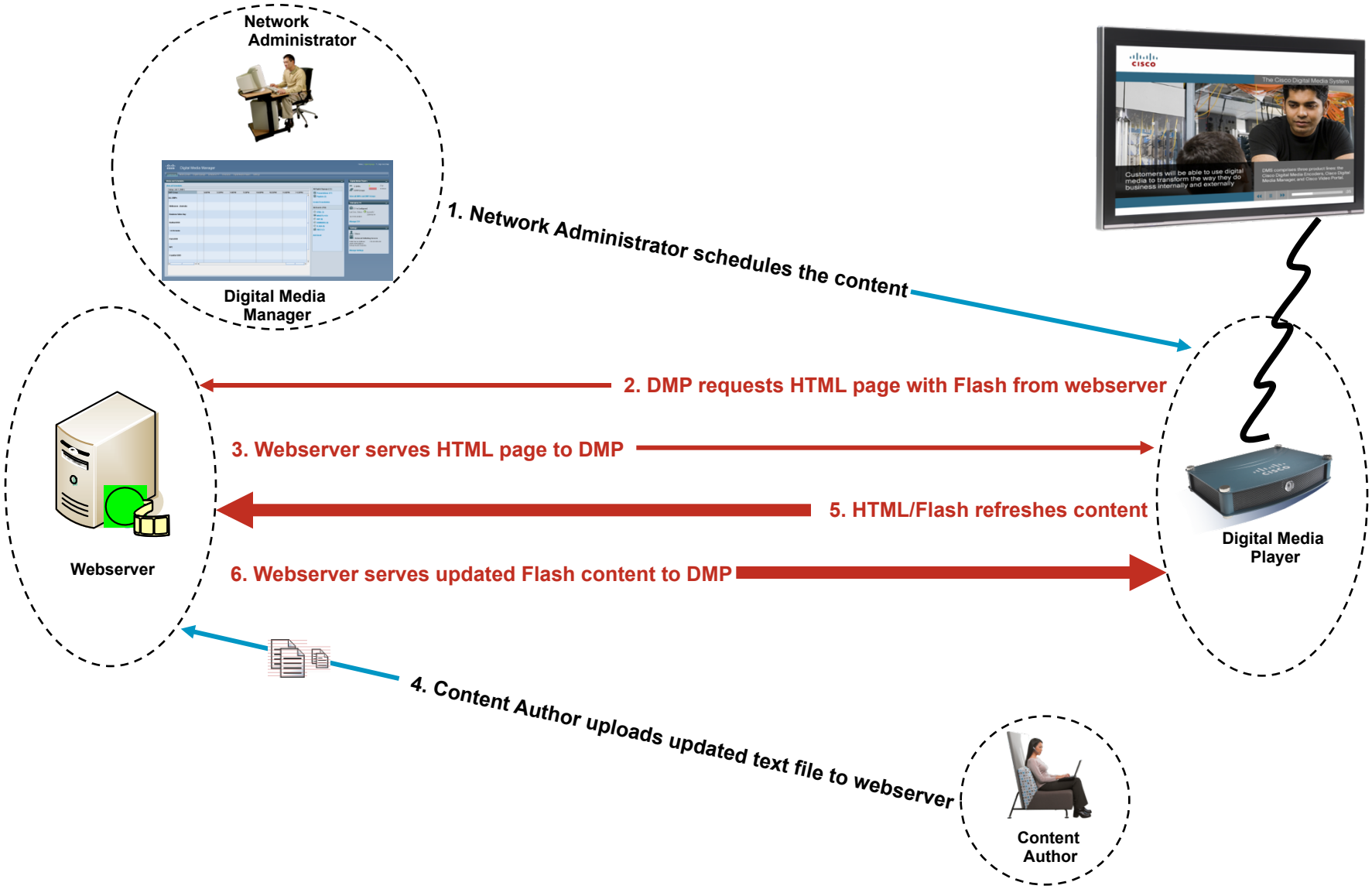
Managing Content outside DMM

Return to Contents

Managing content changes through PPT



Managing content changes through Text/XML files



Dynamic content sample files

META for Automatic Refreshing and Forwarding

code to refresh content page:

<META HTTP-EQUIV="REFRESH" CONTENT="5">

This tells the browser to refresh the page (HTTP -EQUIV="REFRESH"), and that it should do so every five seconds (CONTENT="5").

code to redirect to another page:

<META http-equiv="refresh" content="190; URL=http://Specify URL location here">

This on screen presentation was created in Power Point and converted into flash

Dynamic Content in Web Pages with PHP

- *Hypertext Preprocessor* (PHP) programming language was used in this demo design.
- PHP grabs the data from the text file and creates a Web page as output. Javascript and CSS are also used to render the web page to look as it does in the screen shot.

ARRIVALS

FLIGHT	GATE	SCHED	STATUS	ARRIVING FROM
218	A6	8:40	IN-RANGE	LOS ANGELES
228	A1	8:50	ON TIME	SAN DIEGO
248	B2	9:00	ON TIME	LAS VEGAS
217	D3	8:10	ON TIME	PHOENIX
219	C4	8:40	ON TIME	SAN DIEGO
221	B3	8:50	ON TIME	AUSTIN
522	A9	9:10	ON TIME	LAS VEGAS
231	A6	9:20	ON TIME	LOS ANGELES
720	A11	9:30	ON TIME	SEATTLE
222	A7	9:30	ON TIME	BURBANK
593	C5	9:40	ON TIME	SPOKANE
756	B6	9:40	ON TIME	LOS ANGELES
828	A2	9:50	ON TIME	LOS ANGELES
429	B1	9:50	ON TIME	LOS ANGELES
838	C2	9:50	ON TIME	LOS ANGELES
729	A1	9:50	ON TIME	PORTLAND
398	A4	11:50	ON TIME	LOS ANGELES
269	A3	11:50	ON TIME	SEATTLE

DEPARTURES

FLIGHT	GATE	SCHED	STATUS	DEPARTING TO
332	C2	10:00	ON TIME	LOS ANGELES
788	B1	11:40	ON TIME	LOS ANGELES
728	A1	10:10	ON TIME	SAN DIEGO
948	B2	10:10	ON TIME	LAS VEGAS
517	D3	10:20	ON TIME	PHOENIX
319	C4	10:30	ON TIME	SAN DIEGO
771	B3	10:40	ON TIME	AUSTIN
322	A9	10:50	ON TIME	LAS VEGAS
661	A6	11:00	ON TIME	LOS ANGELES
518	A12	11:00	ON TIME	SEATTLE
717	A7	11:10	ON TIME	BURBANK
516	C5	11:20	ON TIME	SPOKANE
312	B6	11:30	ON TIME	LOS ANGELES
337	A2	11:40	ON TIME	LOS ANGELES
234	D6	10:10	ON TIME	LOS ANGELES
928	C2	11:50	ON TIME	ONTARIO
798	D6	11:50	ON TIME	ORLANDO
437	B2	11:40	ON TIME	LOS ANGELES

9/25/2006 6:35:17 AM

rates more than 3,400 flights a day on United, United Express® and TedSM to more than 200 U.S. domestic and

PHP for Dynamically Changing Content

- PHP pages and txt files

flights.dat - Notepad

File Edit Format View Help

```
# Flight|Gate|Sched|Status|From|Flight|Gate|Sched|Status|To□218|A6|
n time|ontario□398|a4|11:50|on time|los angeles|798|d6|11:50|on tim
```

tvUtils.php | tvTemplate.php* | tvInit.php | tvConsts.php | index.html

```php
<?php

function tvPageBegin($bodyTemplate, $headerTemplate = 'header.tpl', $footerTemplate = 'footer.tpl')
{
  global $tvTemplate;

  $tvTemplate->set_file(array(
    'HEADER' => $headerTemplate,
    'BODY' => $bodyTemplate,
    'FOOTER' => $footerTemplate
  ));
}

function tvPageEnd()
{
  global $tvTemplate;

  $tvTemplate->parse('HEADER', 'HEADER');
  $tvTemplate->parse('FOOTER', 'FOOTER');
  $tvTemplate->parse('BODY', 'BODY');

  $tvTemplate->p('BODY');
}

function tvGetFieldValue($name)
{
  if (array_key_exists($name, $_POST))
    return $_POST[$name];
  else
    if (array_key_exists($name, $_GET))
      return $_GET[$name];

  return NULL;
}

?>
```

Getting field and name values

Using Post and Get Methods

HTML for Dynamically Changing Content

- HTML page (fragment) that was rendered as a result of PHP programming

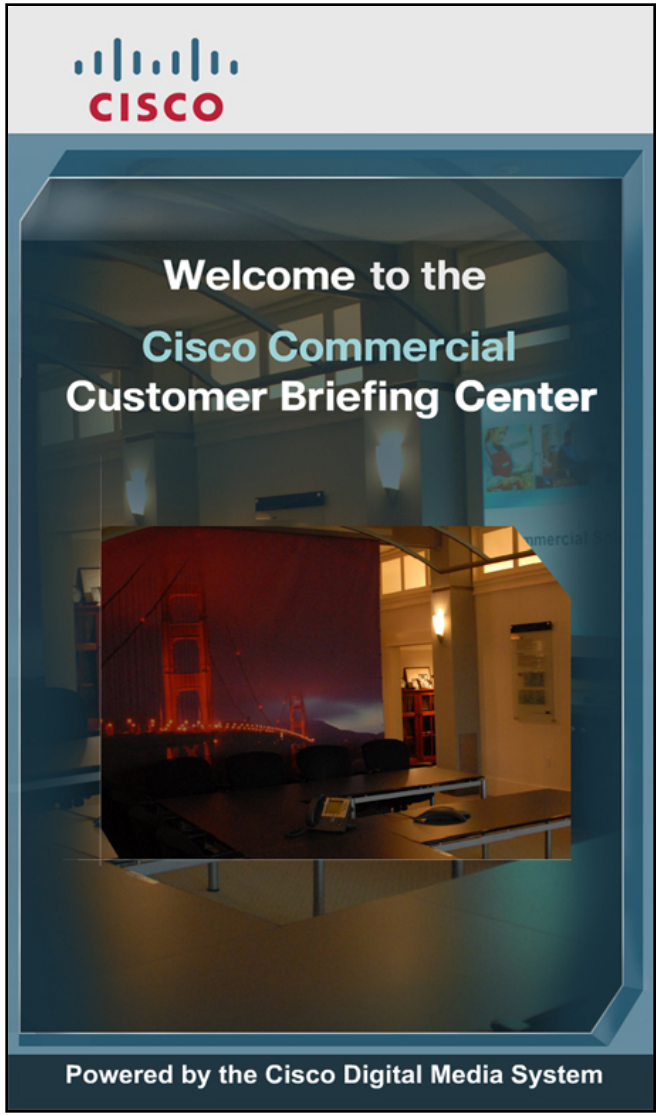
```
<META content="MSHTML 6.00.2900.2976" name=GENERATOR></HEAD>
<BODY onload=init();>
<TABLE class=schedule cellSpacing=0 cellPadding=0 border=0>
  <TBODY>
  <TR style="HEIGHT: 83px">
    <TD colSpan=10> </TD></TR>
  <TR class=columnCaption>
    <TD width=130>FLIGHT</TD>
    <TD width=82>GATE</TD>
    <TD width=96>SCHED</TD>
    <TD width=138>STATUS</TD>
    <TD width=230>ARRIVING FROM</TD>
    <TD width=130>FLIGHT</TD>
    <TD width=82>GATE</TD>
    <TD width=96>SCHED</TD>
    <TD width=138>STATUS</TD>
    <TD width=244>DEPARTING TO</TD></TR>
  <TR style="HEIGHT: 35px">
    <TD colSpan=10> </TD></TR>
  <TR class=row1>
    <TD id=aflight0>218</TD>
    <TD id=agate0>A6</TD>
    <TD id=asched0>8:40</TD>
    <TD id=astatus0 style="PADDING-LEFT: 10px; TEXT-ALIGN: left">IN-RANGE</TD>
    <TD id=acity0 style="PADDING-LEFT: 10px; TEXT-ALIGN: left">LOS ANGELES</TD>
    <TD id=dflight0>332</TD>
    <TD id=dgate0>C2</TD>
    <TD id=dsched0>10:00</TD>
    <TD id=dstatus0 style="PADDING-LEFT: 10px; TEXT-ALIGN: left">ON TIME</TD>
    <TD id=dcity0 style="PADDING-LEFT: 10px; TEXT-ALIGN: left">LOS
  ANGELES</TD></TR>
  <TR class=row2>
    <TD id=aflight1>228</TD>
```

Examples of Stand-Alone Templates

- This file was created outside of the DMM.
- This template and its content can be hosted on a web and/or video server or stored on the DMP for local playback.

Horizontal Stand-Alone Sample Designs

Vertical Stand-Alone Sample Designs



Additional Resources:

Conversion: PPT to .swf via PowerConverter

Return to Contents

If you have PowerConverter

Power Point application for PC users or Keynote for Apple users can be used for presentations creation.

For DMP playback it is optional to *convert* presentations into *swf* file format, or *export jpeg images* from Power Point.

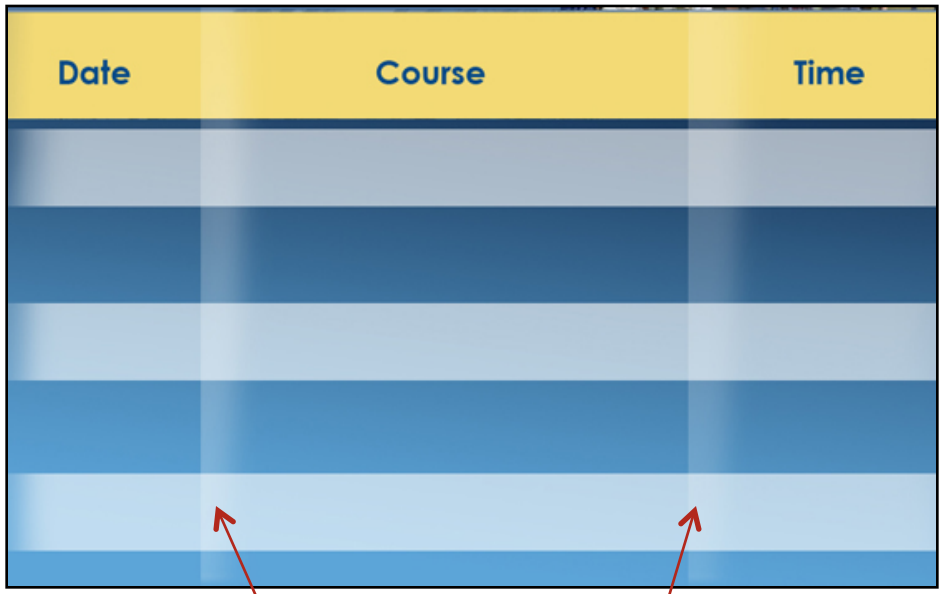
Keep your presentation simple.

If you have PP plug-in such as Power Converter, please follow our recommendations in following slides.

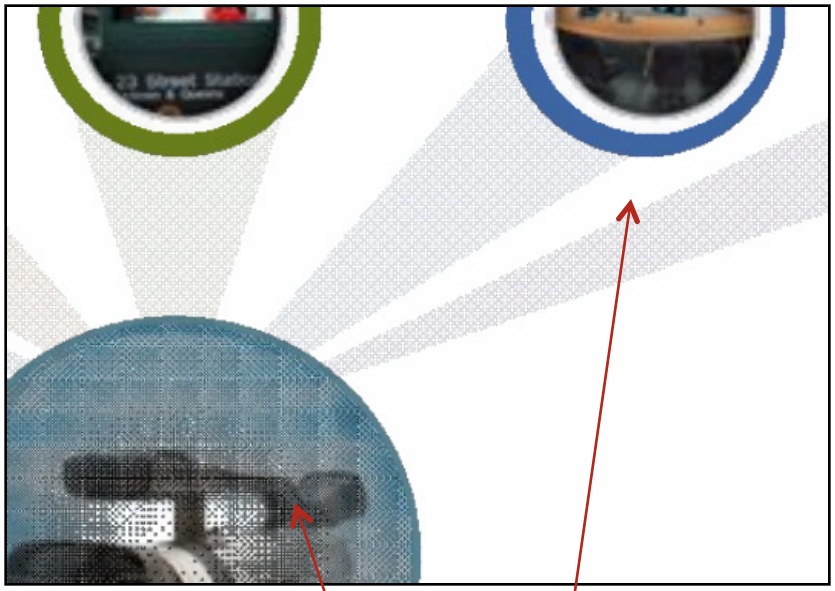
PP to swf conversion using PowerConverter

PP creation for swf conversion

For best results use the images, do not use transparency, shadows, or other effects applied to auto shapes in PP



a Transparency created in Photoshop after PP to swf conversion



b Transparency created in after PP to swf conversion

Coded swf vs. swf Conversion Output

Coded swf application allows direct control of:

- ✓ Data,
- ✓ Text,
- ✓ Images,
- ✓ Slide Duration,
- ✓ Upload
- ✓ etc. including other programming controls

Power Point swf conversion output does not allow you such controls, however some workaround can be done to optimize playback:

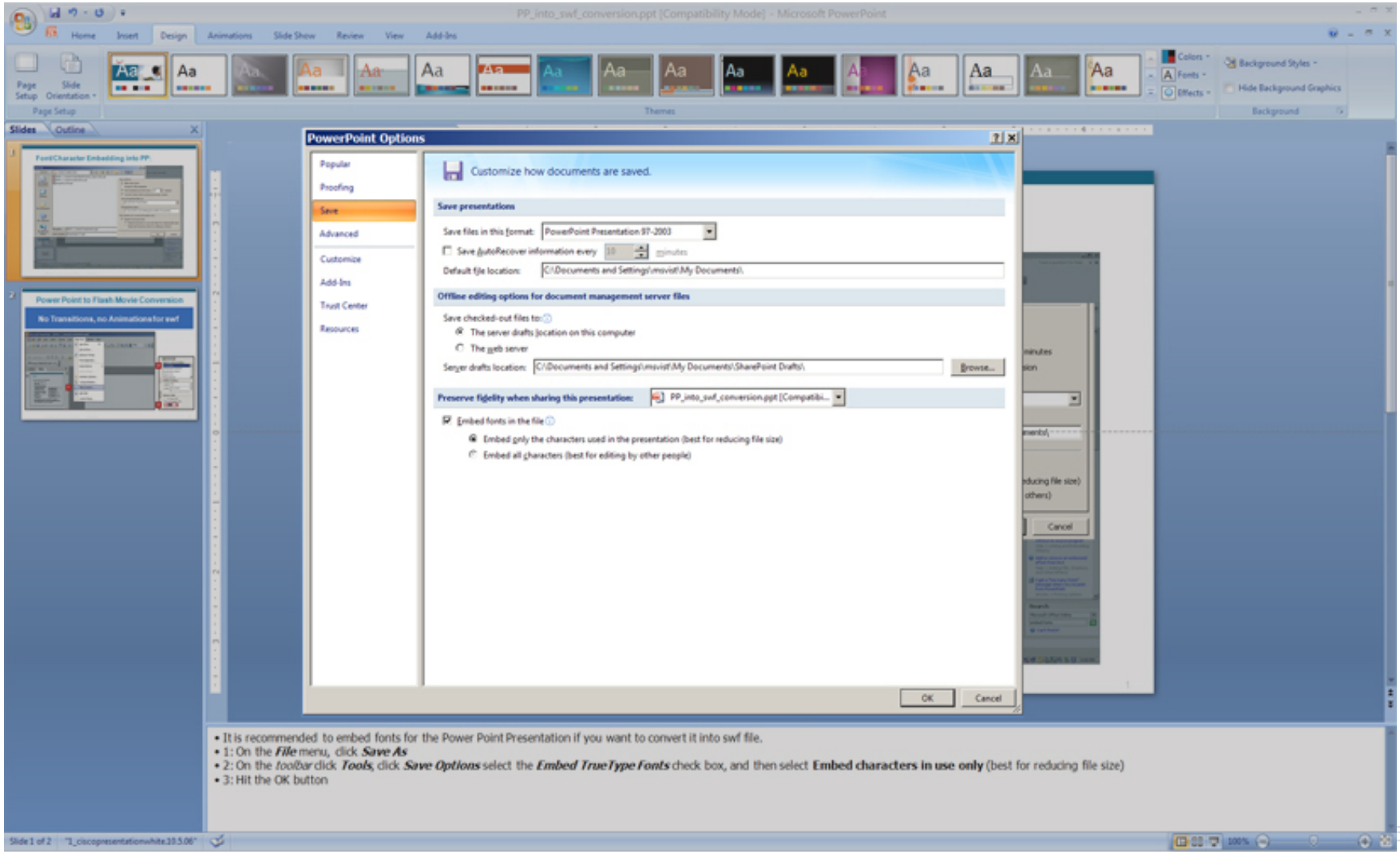
- ✓ Use Java Script to preload presentations
- ✓ Minimize the size of your swf presentation to the frequently updated information

Coded swf application example

Swf conversion outputs example

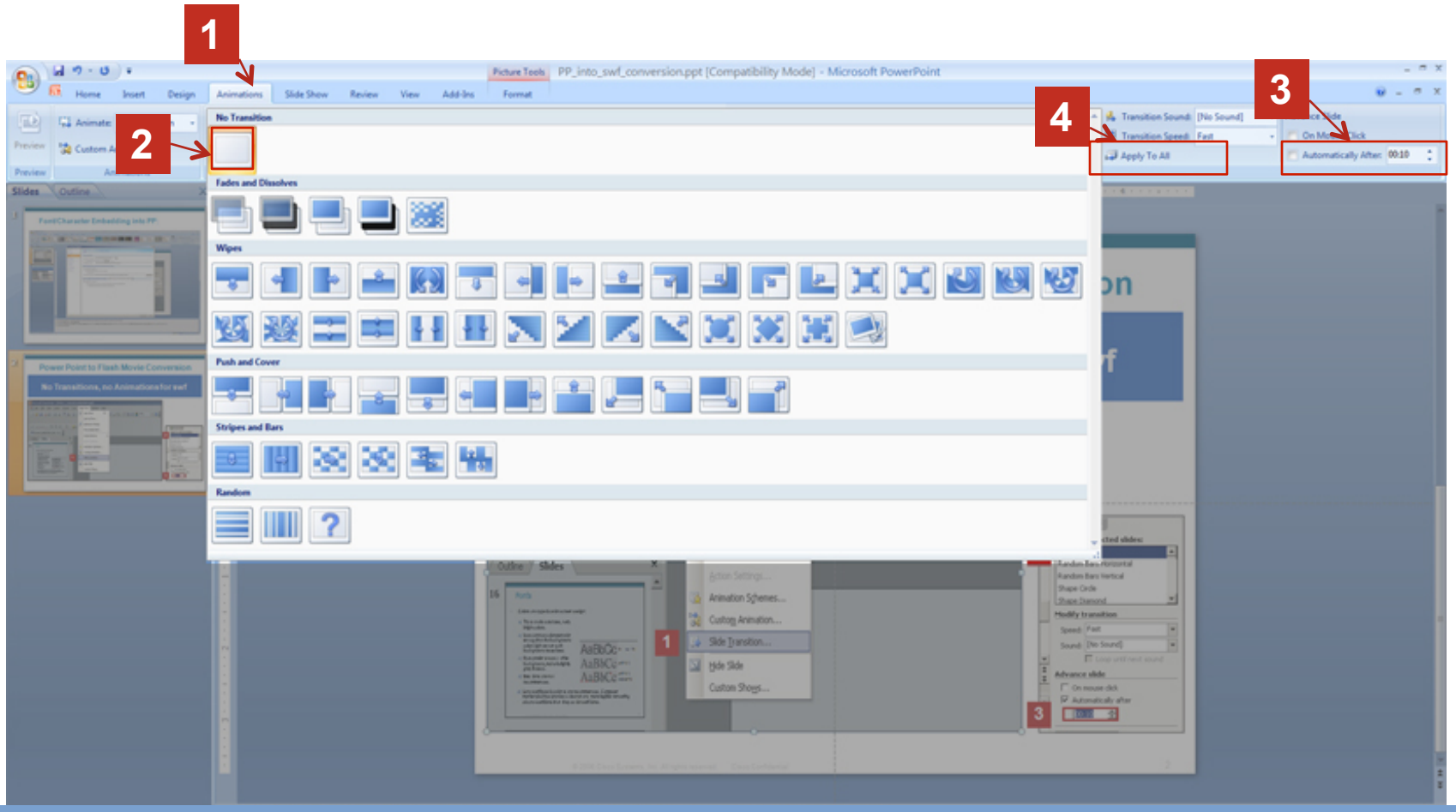
PP to swf conversion using PowerConverter

Font/Character Embedding into PP:



PP to swf conversion using PowerConverter

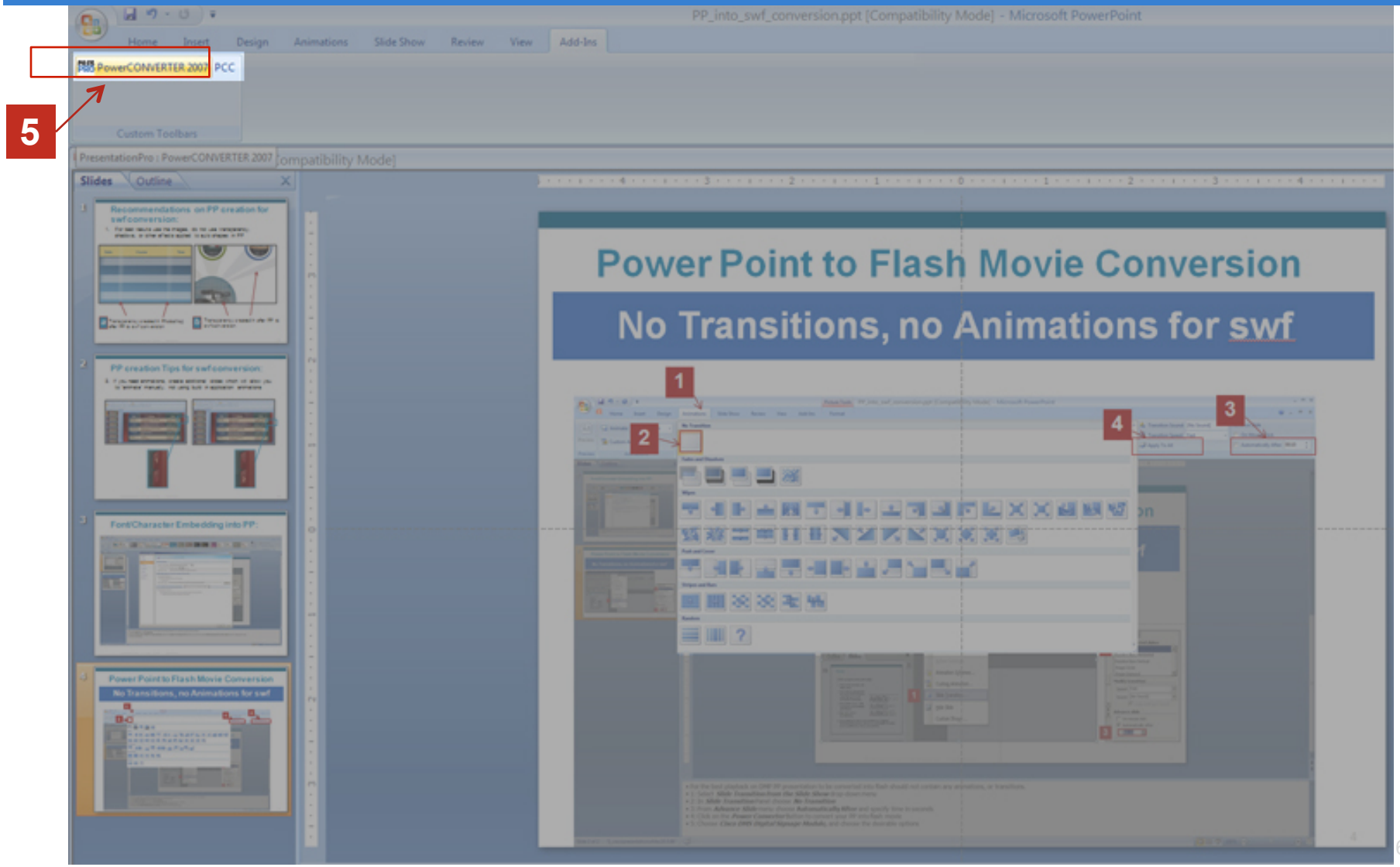
Step-by-step Instructions



No Transitions, no Animations for swf

PP to swf conversion using PowerConverter

Step-by-step Instructions



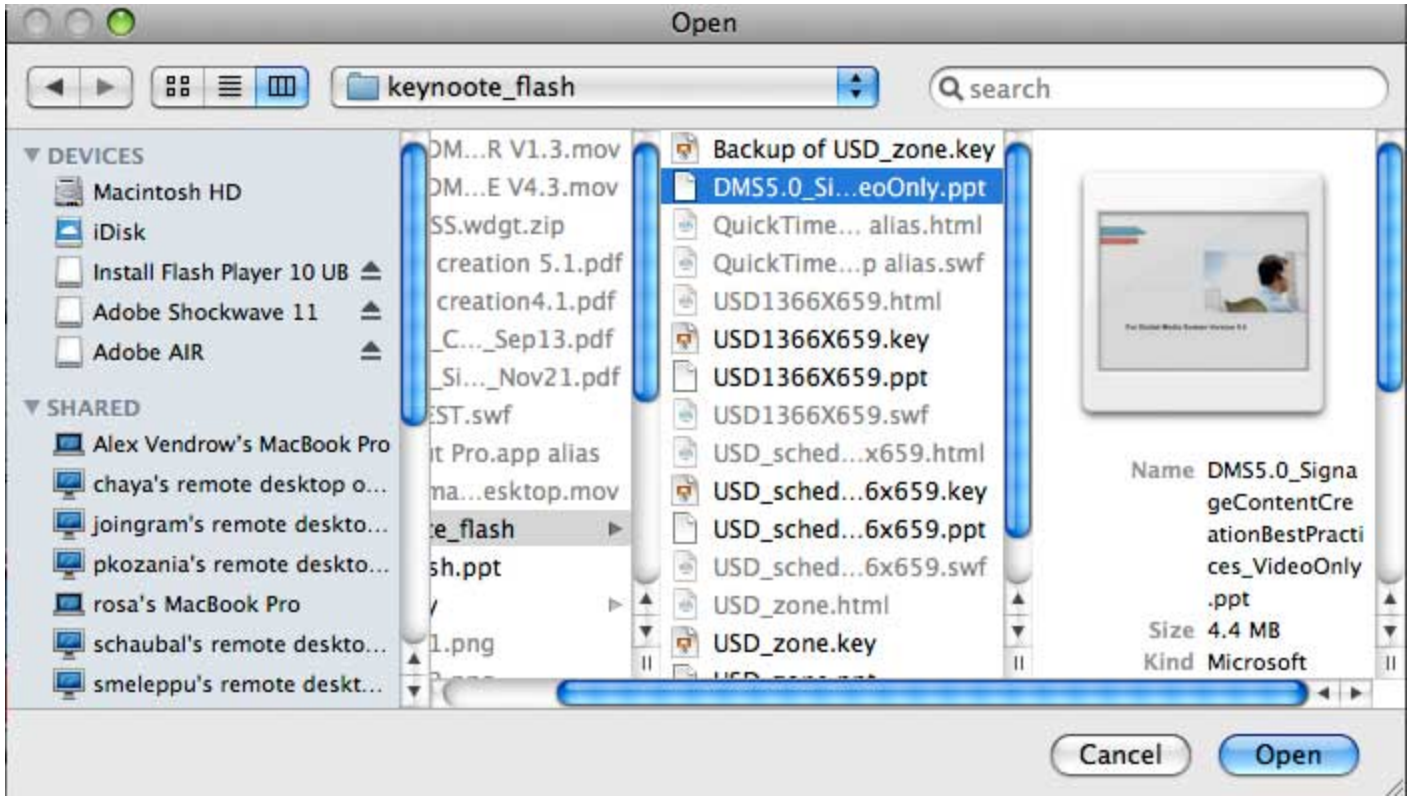
Additional Resources:

Conversion: PPT to .swf via Apple Keynote.

Return to Contents

Apple KeyNote: PPT to .swf export

- ***Step 1*: Open a PowerPoint file within Keynote**



Apple KeyNote: PPT to .swf export

- *Step 2: Prepare the timing between slides.*

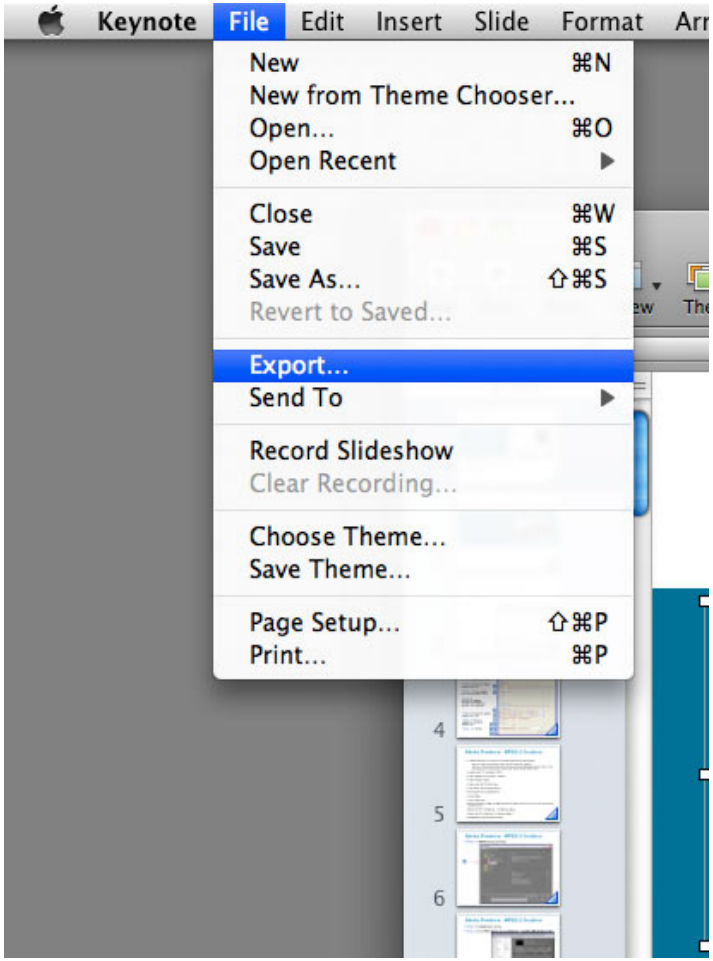
While Keynote generates a very legible .swf, transitions will not export to the .swf be it a PowerPoint of native Keynote file.

Apple KeyNote: PPT to .swf export

- *Step 3: Once configured, save your work: File>Save as…*

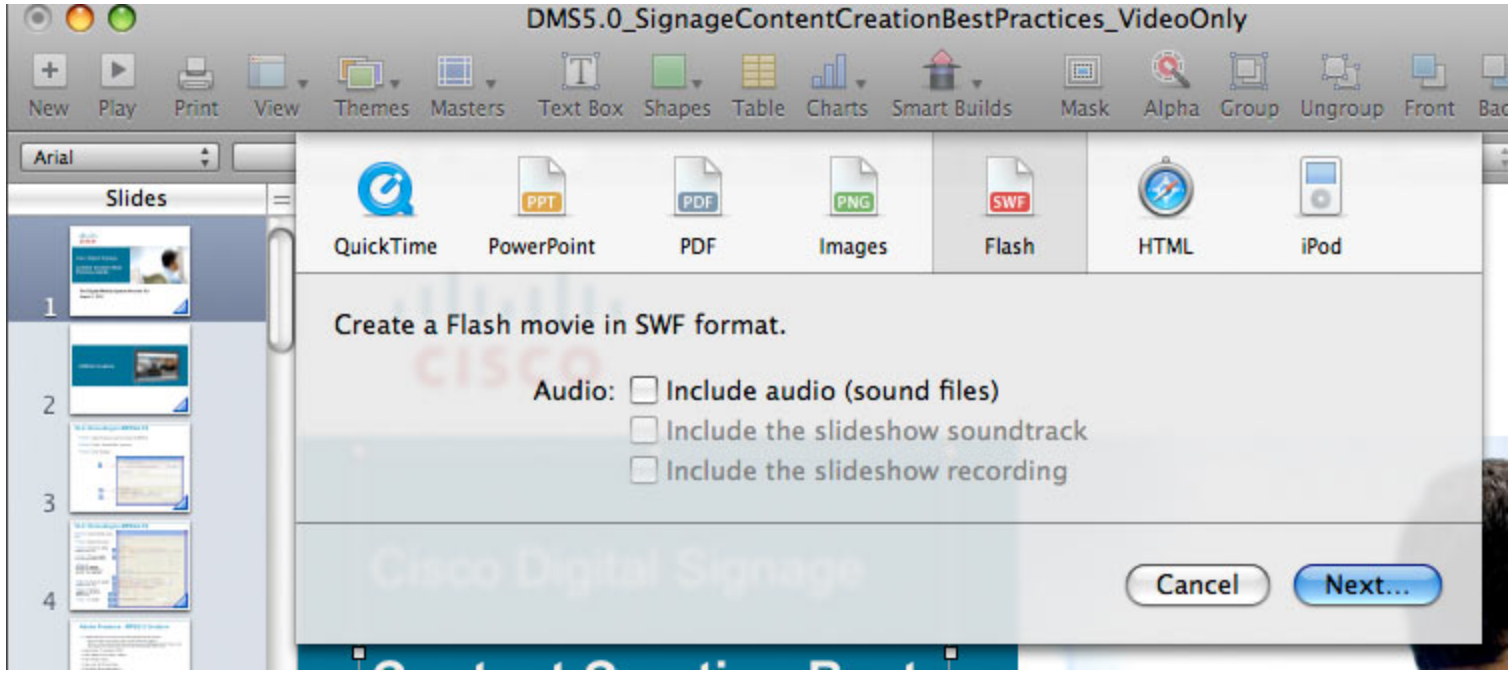
Apple KeyNote: PPT to .swf export

- ***Step 4*: To create a Flash .swf use: File > Export**



Apple KeyNote: PPT to .swf export

- ***Step 5*: Click the Flash button, name your .swf, and test once rendered.**



Additional Resources:

Conversion: .swf to mpeg via Moyea

Return to Contents

Swf to mpeg conversion using Adobe Premiere or Adobe After Effects.

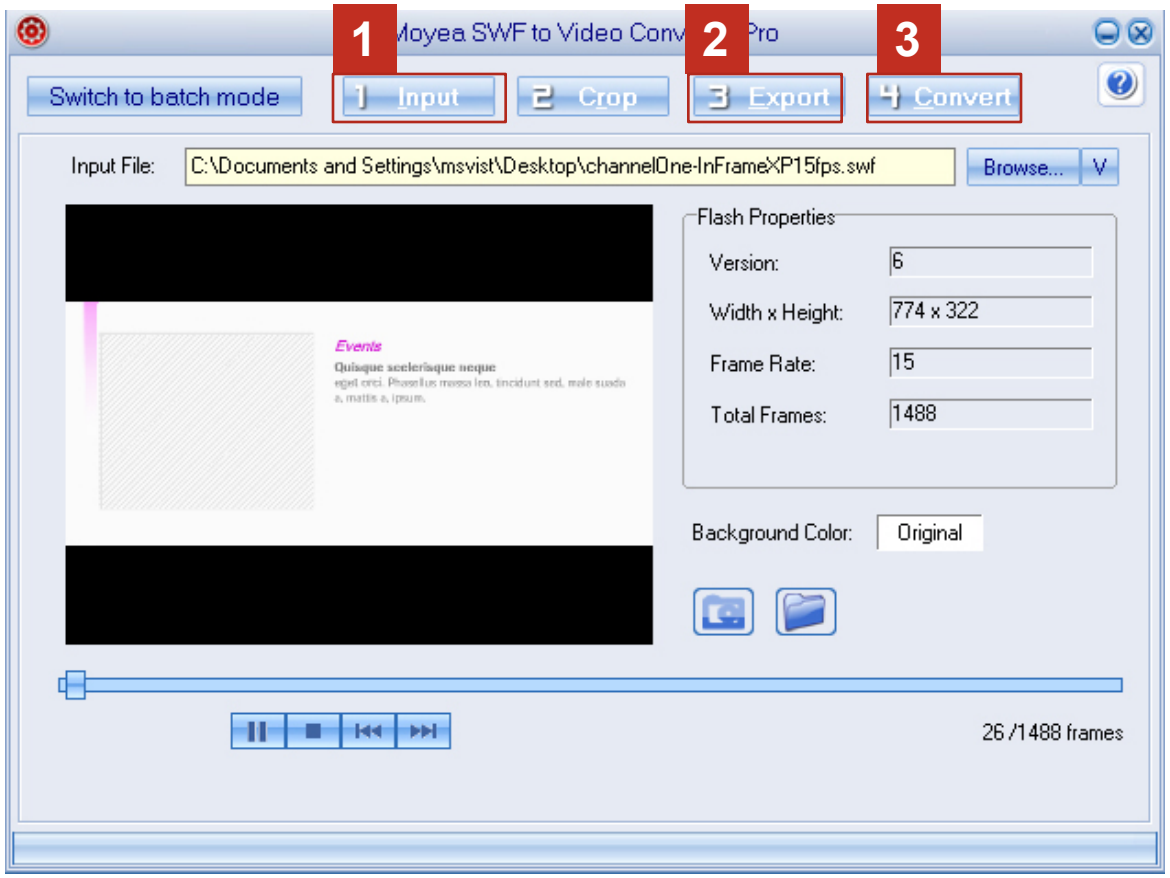
Preferred way of swf to mpeg conversion

1. Frame rate in flash file needs to be set to 24fpc-29.97fpc
2. Flash needs to be same size as desired video frame size
3. Swf needs to be saved.
4. In new flash document saved swf needs to be imported to the stage, not to the library.
5. Re-imported to stage animation needs to be exported as image sequence.
6. Image sequence needs to be imported to video editing application on your choice.

Note: This conversion produces best results.

swf to mpg conversion using Moyea

Alternative way of swf to mpeg conversion

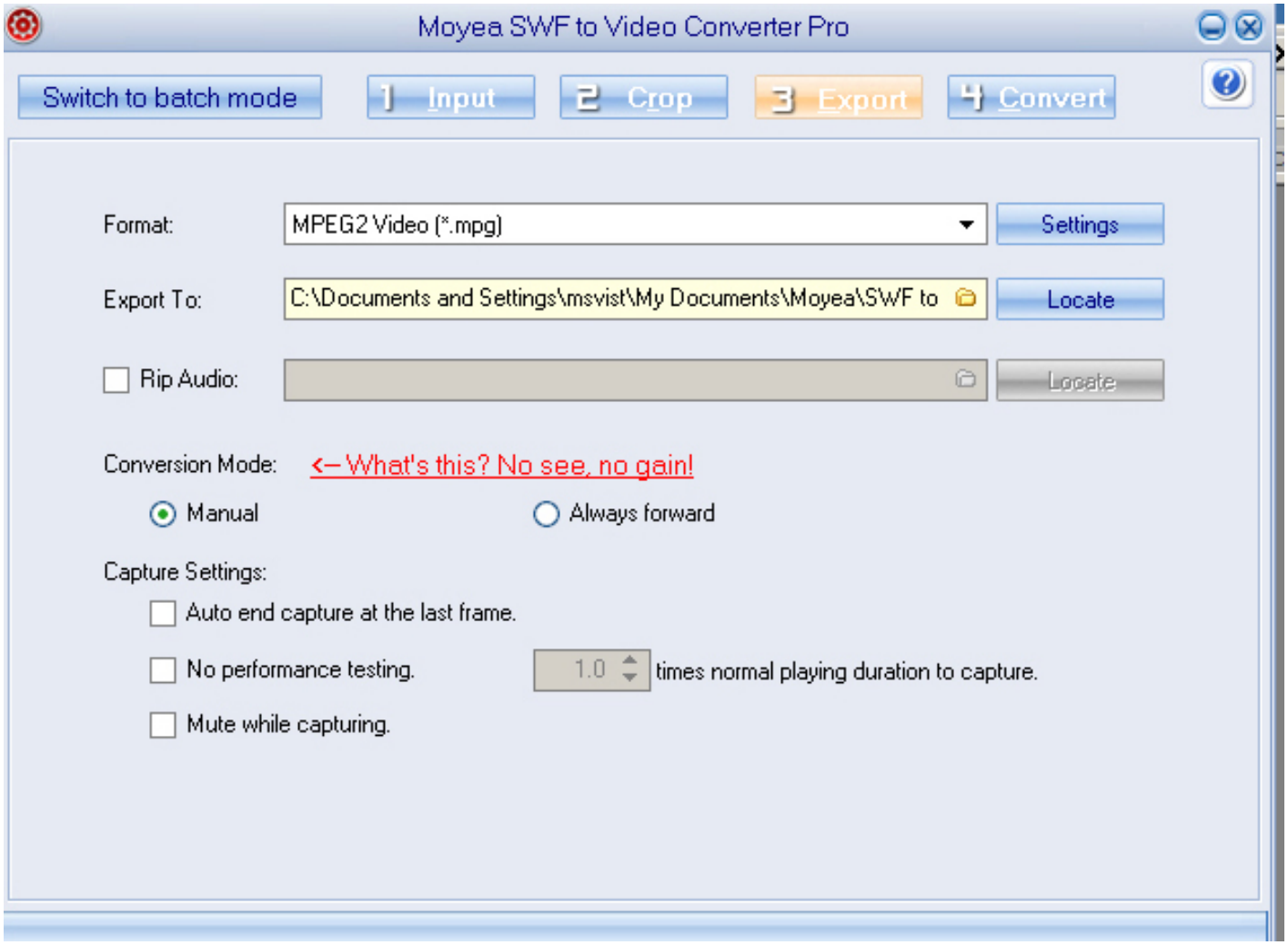


1. Click Input button to locate file to be converted
2. Click Export button to choose from formats drop down list mpeg2
3. Click the Convert button to convert the movie

If you wish to apply h.264 codec to final mpeg, use VLC

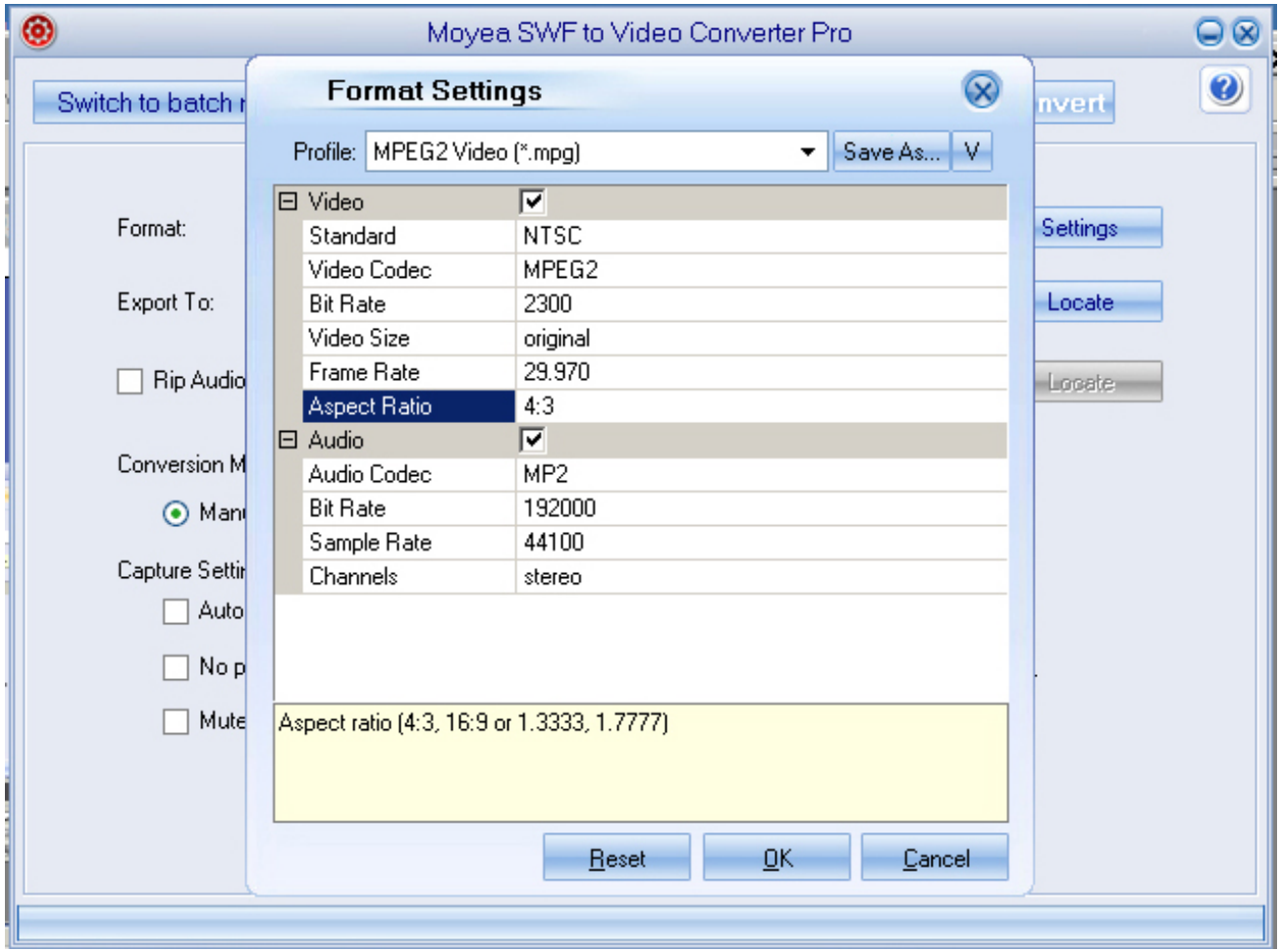
swf to mpg conversion using Moyea

Moyea Interface. Export Panel



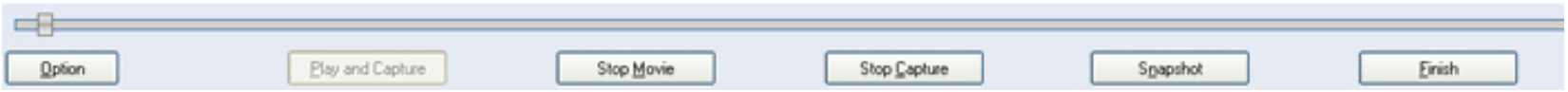
swf to mpg conversion using Moyea

Moyea Interface. Export Settings



swf to mpg conversion using Moyea

Moyea Interface. Play and Capture Mode



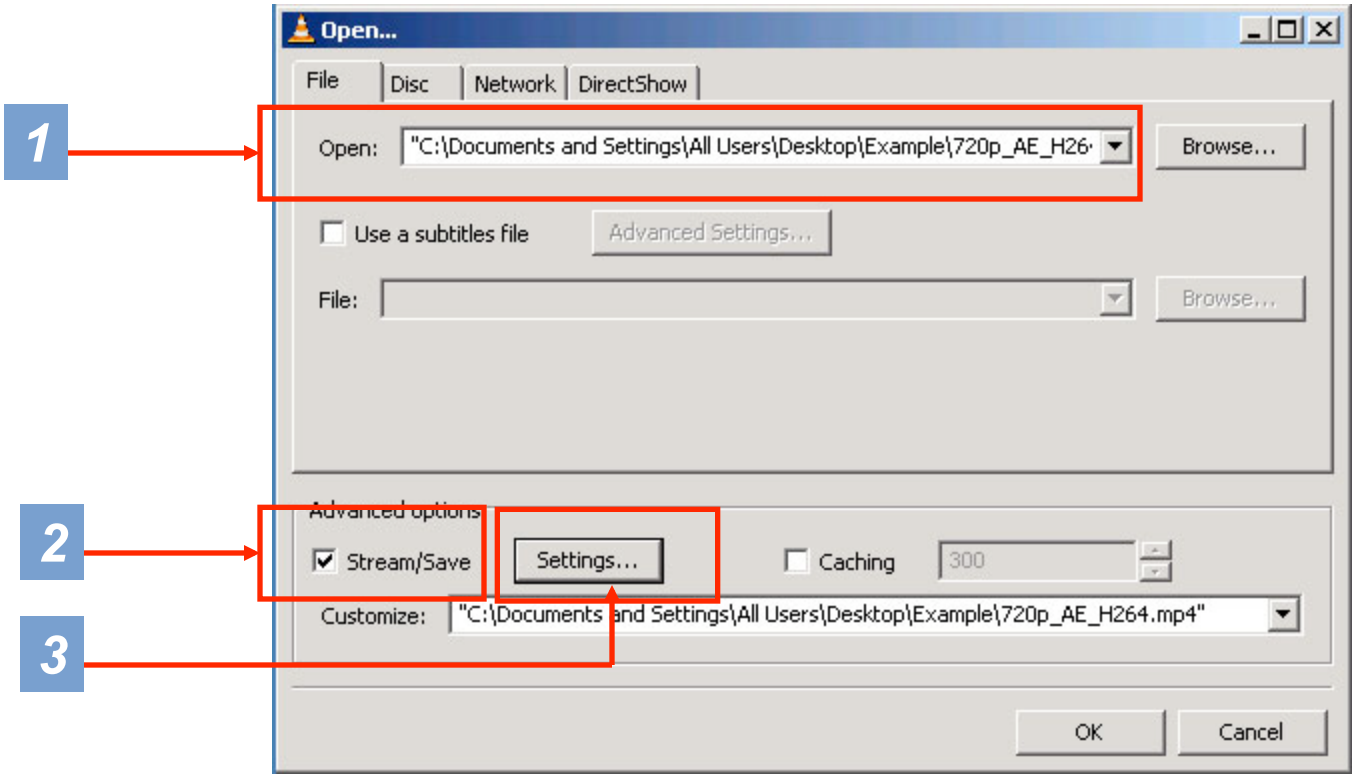
Additional Resources:

Reformatting:
MPEG4 to Mpeg2 –TS

Return to Contents

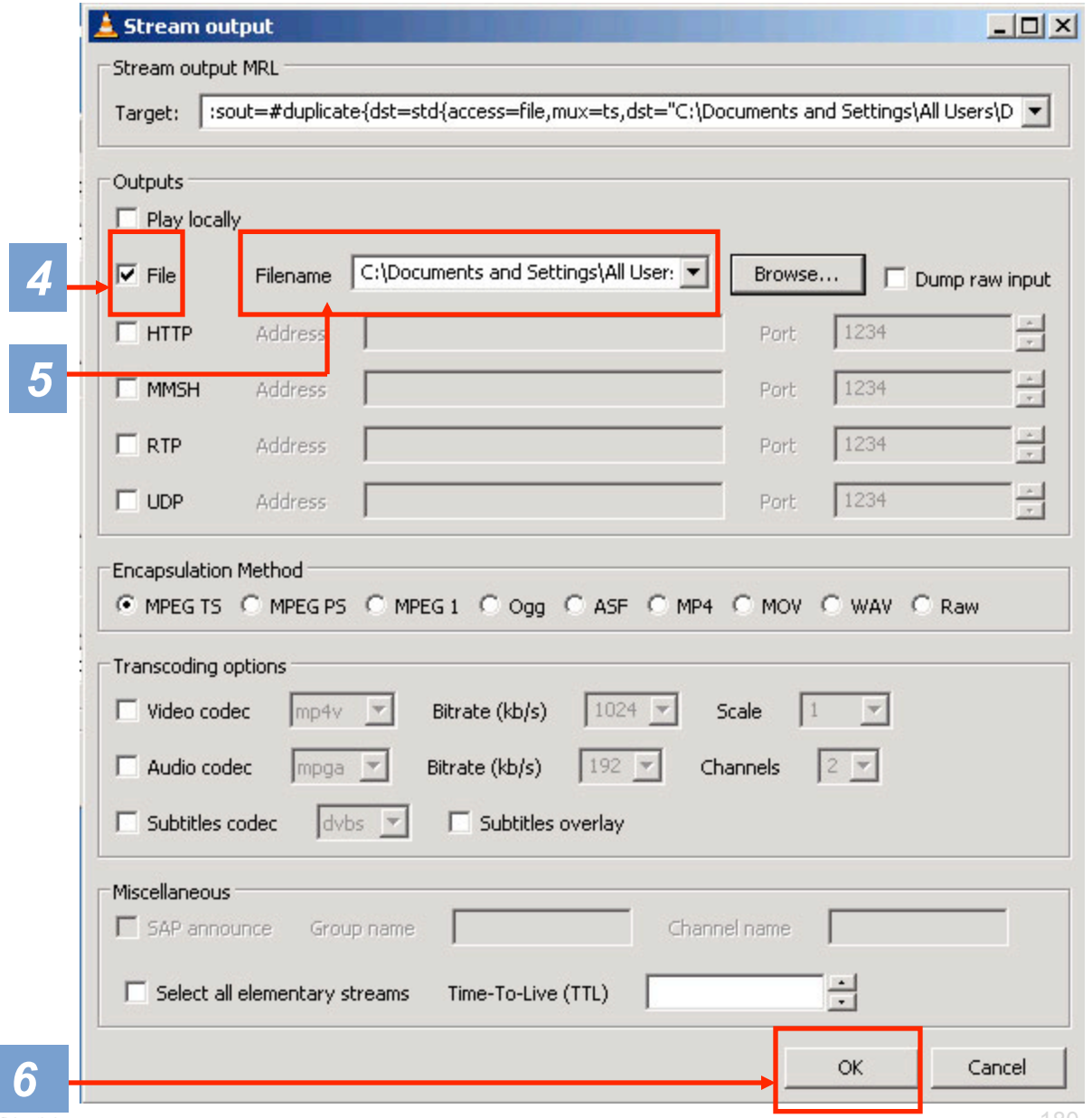
VLC: MPEG4 to Mpeg2 –TS

- ***Step 1:*** Open the file you need to re-format (wrap) into MPEG2-TS.
- ***Step 2:*** Check "Stream/Save" check-box
- ***Step 3:*** Click "Settings"



VLC: MPEG4 to Mpeg2 –TS

- ***Step 4:*** Check the ***file*** check box
- ***Step 5:*** Specify file name
- ***Step 6:*** Click “OK”.



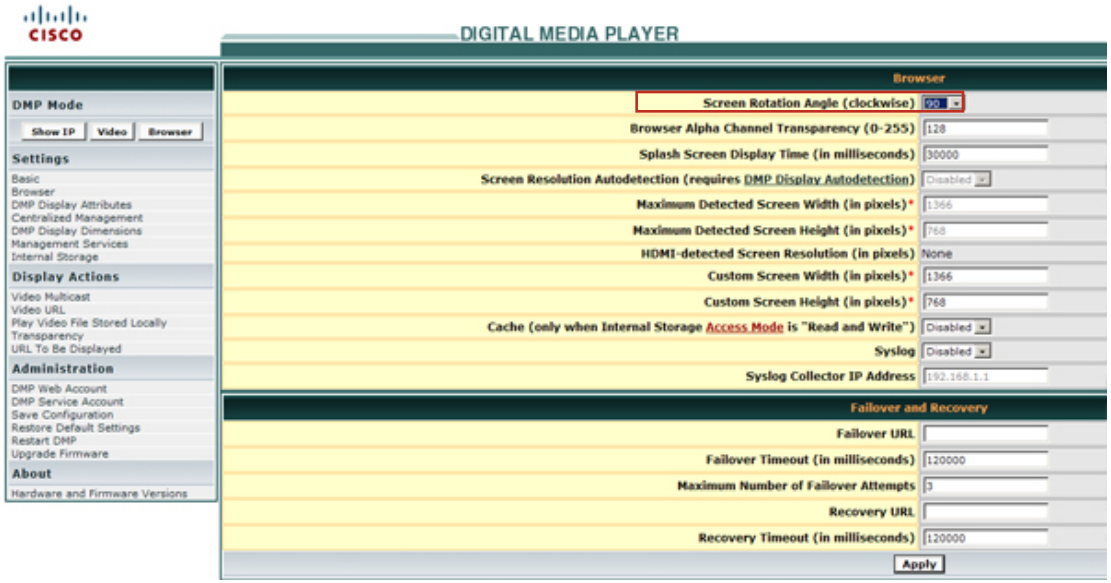
Additional Resources:

Custom Video Framing

Return to Contents

Vertical Videos for DMP Playback

- To play the videos on Vertical Displays you need to use rotated videos and adjust the **rotation in DMP using DMP web interface:**



- ***Step 1*: *From DMP Mode*** menu choose **Browser**

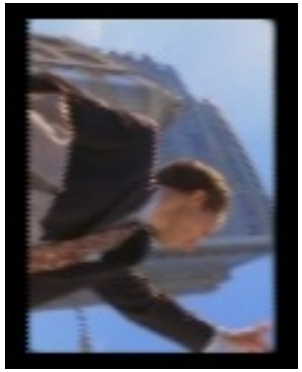
- ***Step 2*: *From <u>Screen Rotation Angle</u>*** <u>drop down menu ***choose 90***</u>

Custom: Video rotation vs. Video framing

- Most non-linear editors programs can rotate a video within the standard 4:3 or 16:9 frame; however, black bars are produced to maintain the horizontal dimension of the frame aspect ratio.
- Custom framing eliminates these black bars; esp., when a video is rotated.
- An exception is the MPEG-1 format, which can produce a borderless vertical format video but at the greater expense of a large file size. Testing such file sizes for playback through the DMP is strongly recommended.

Original Video

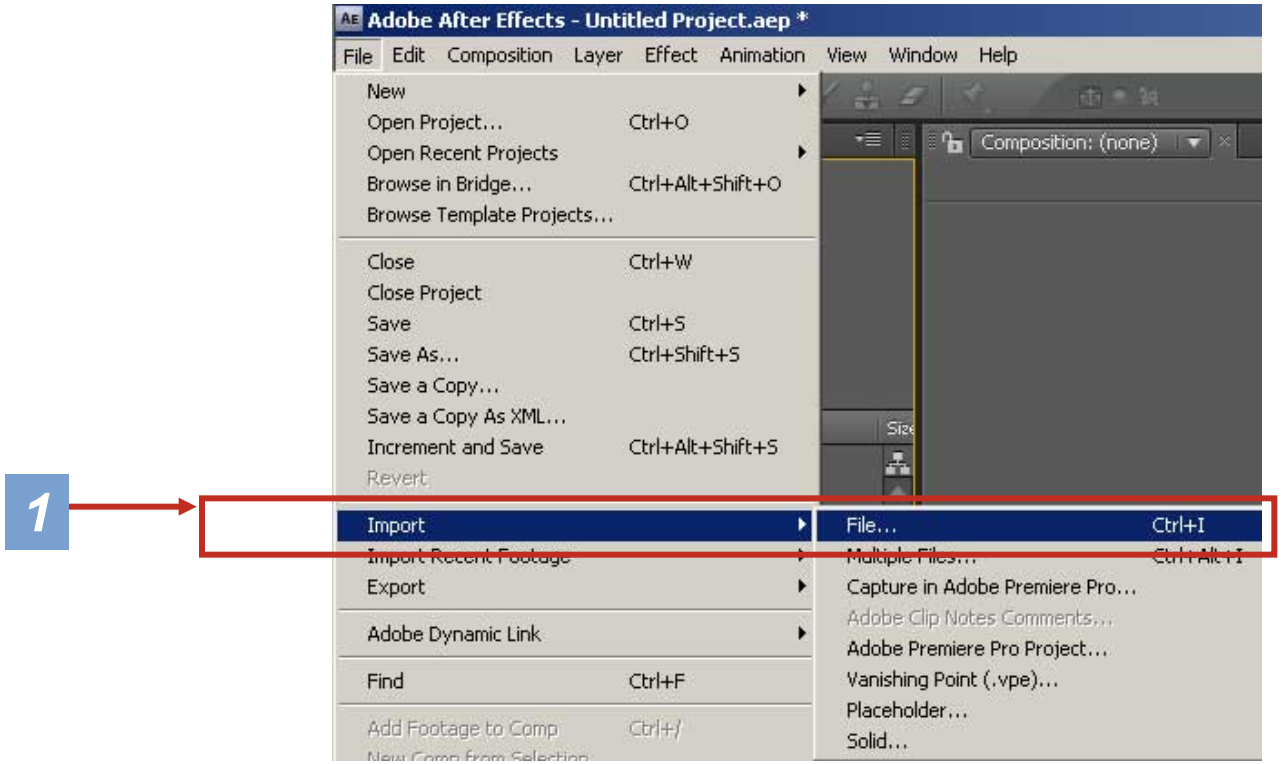
Video rotated 90



Video framed 90

Custom Framing Vertical Video: After Effects CS4

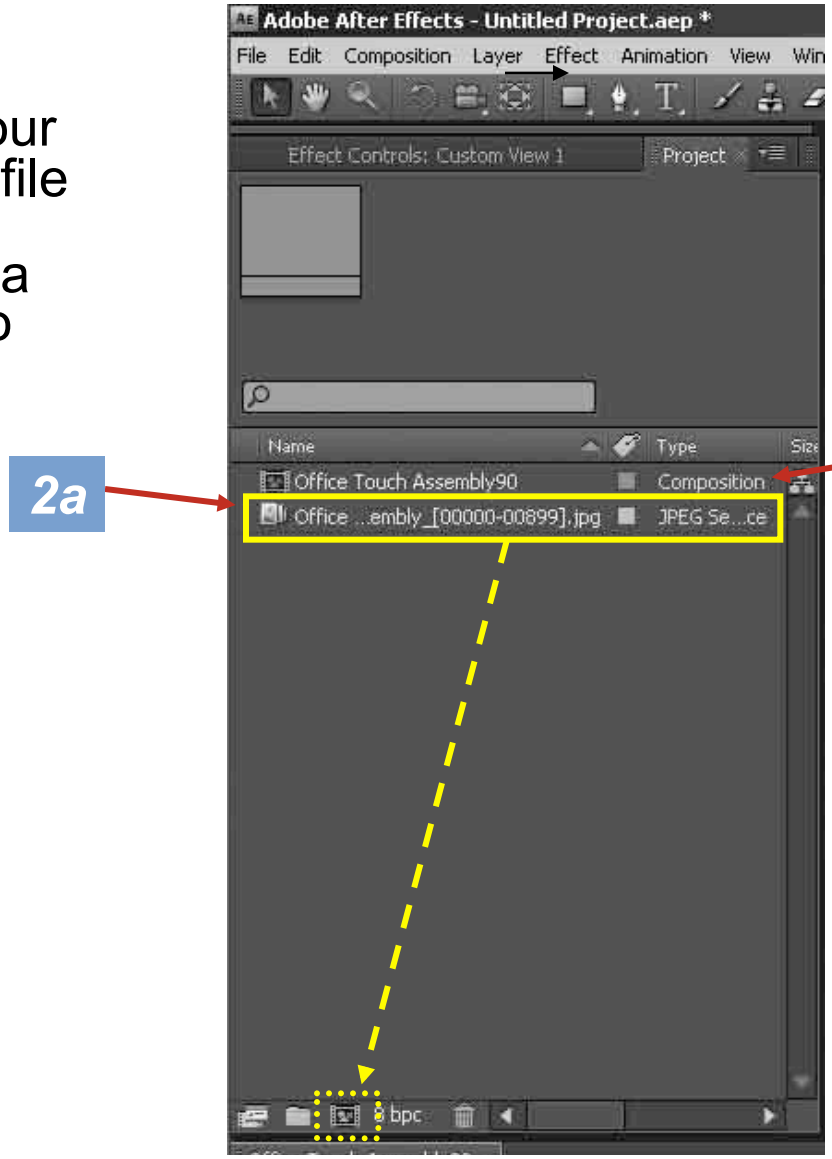
- ***Step 1*: File>Import>file: to load footage into the Project Bin**



- ***Note*: These steps assume a non-interlaced, square pixel, output playing within a template. Where necessary, convert your source footage to a progressive, square pixel state at a maximum screen size of 1366x768.**

Custom Framing Vertical Video: After Effects CS4

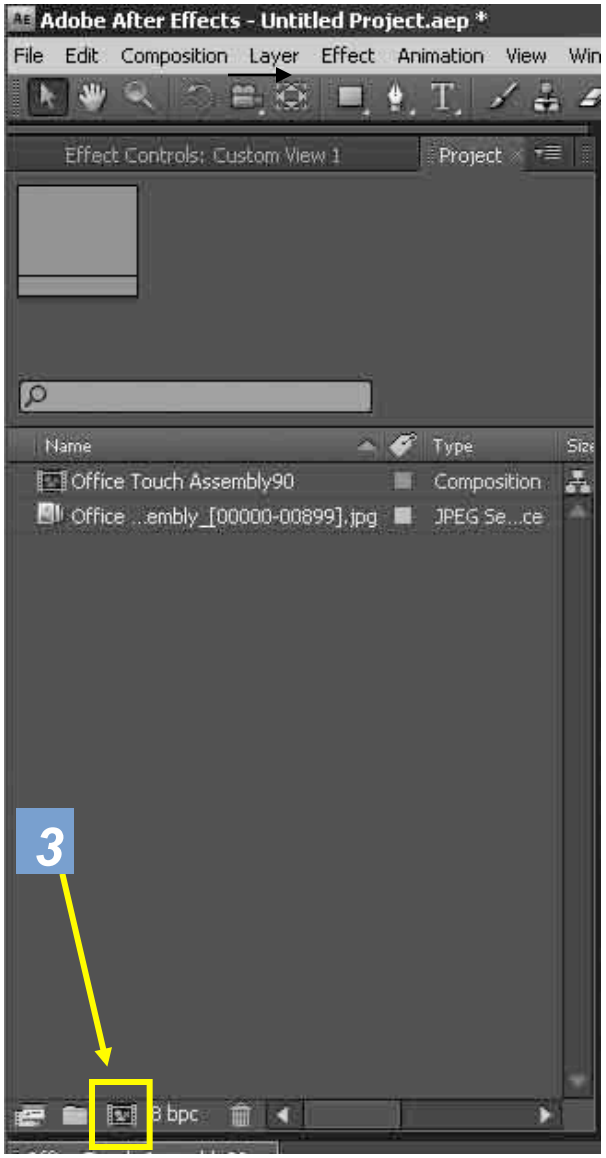
- ***Step 2a/b*: *create*** your composite by dragging file onto comp button to automatically generate a composition matched to your footage.



The new composition will then appear in the Project bin

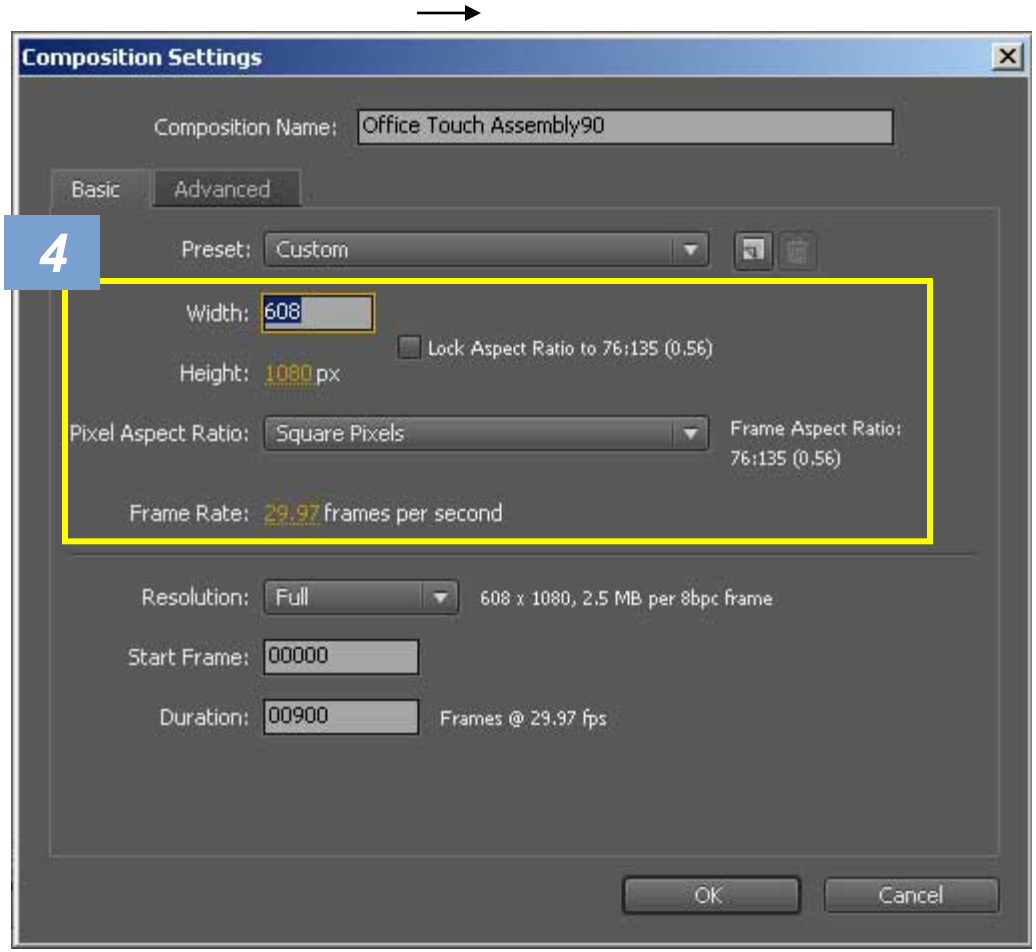
Custom Framing Vertical Video: After Effects CS4

- ***Step 3*: *create*** a new composite by directly clicking the comp button to generate a separate composition



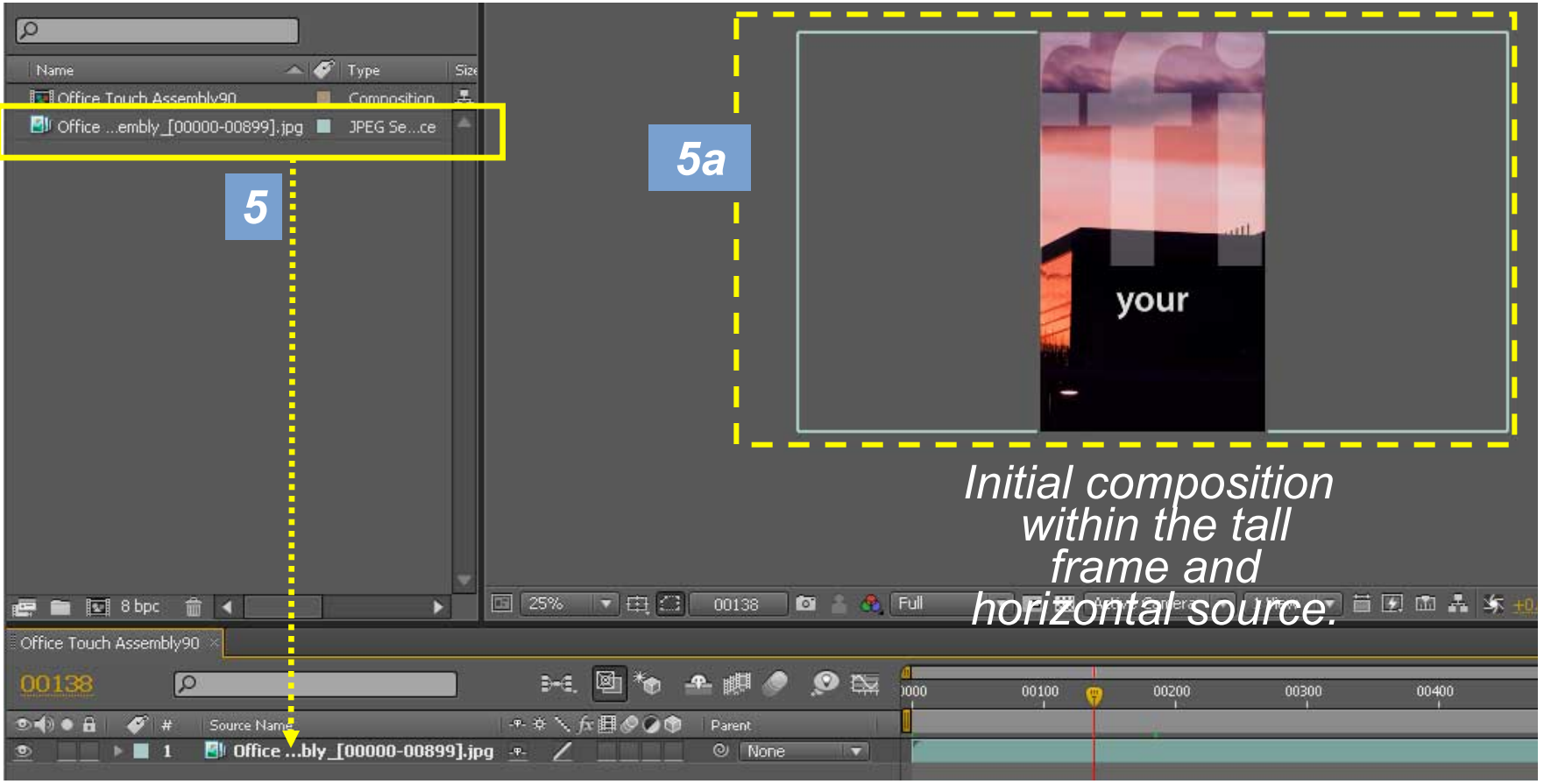
Custom Framing Vertical Video: After Effects CS4

- ***Step 4:*** In the open "Composition Settings" dialogue box:
- Uncheck "Lock Aspect Ratio" and set the frame dimensions to the desired vertical dimensions.
- Click "OK".



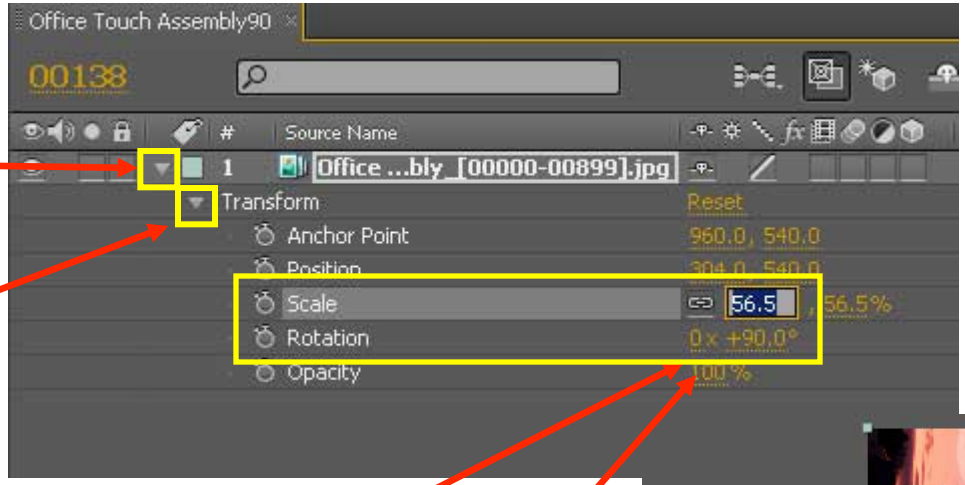
Custom Framing Vertical Video: After Effects CS4

- ***Step 5, 5a*: *drag*** your first composition into the empty composition panel, to nest it within the “tall” format.



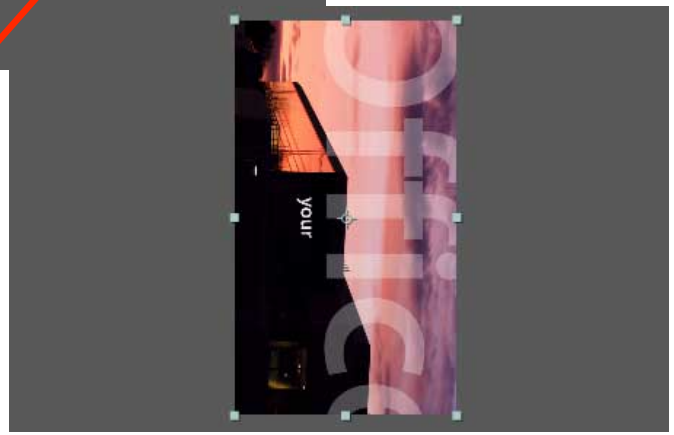
Custom Framing Vertical Video: After Effects CS4

- ***Step 6:***
  Click the twirl down arrow to the left of the layer name, then the twirl down next to "Transform".



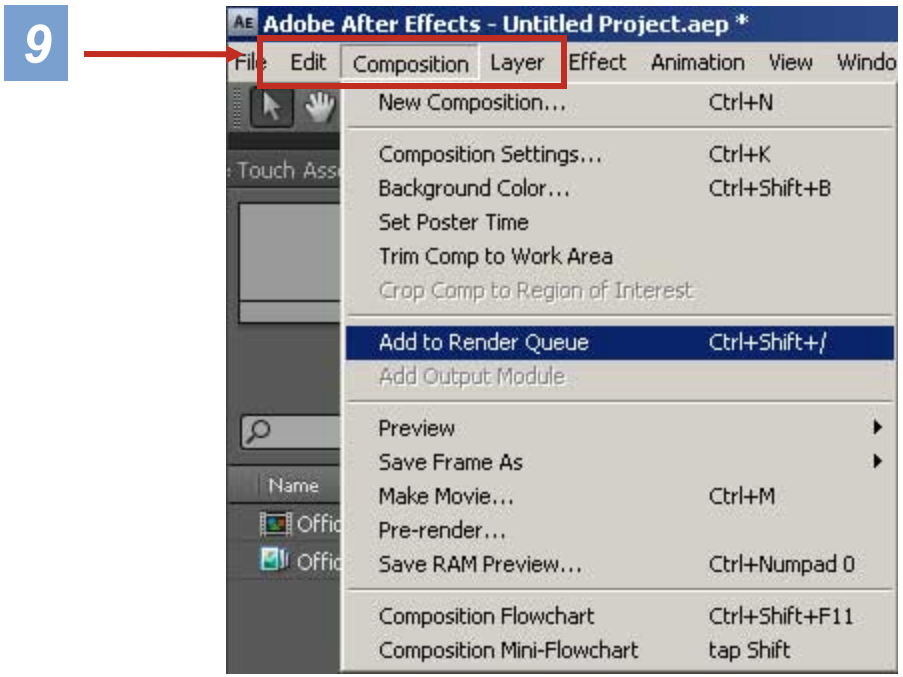
- ***Step 7: Use*** a value of "90" or "270" for *"Rotation".*

- ***Step 8: Manually set*** a value for "Scale".



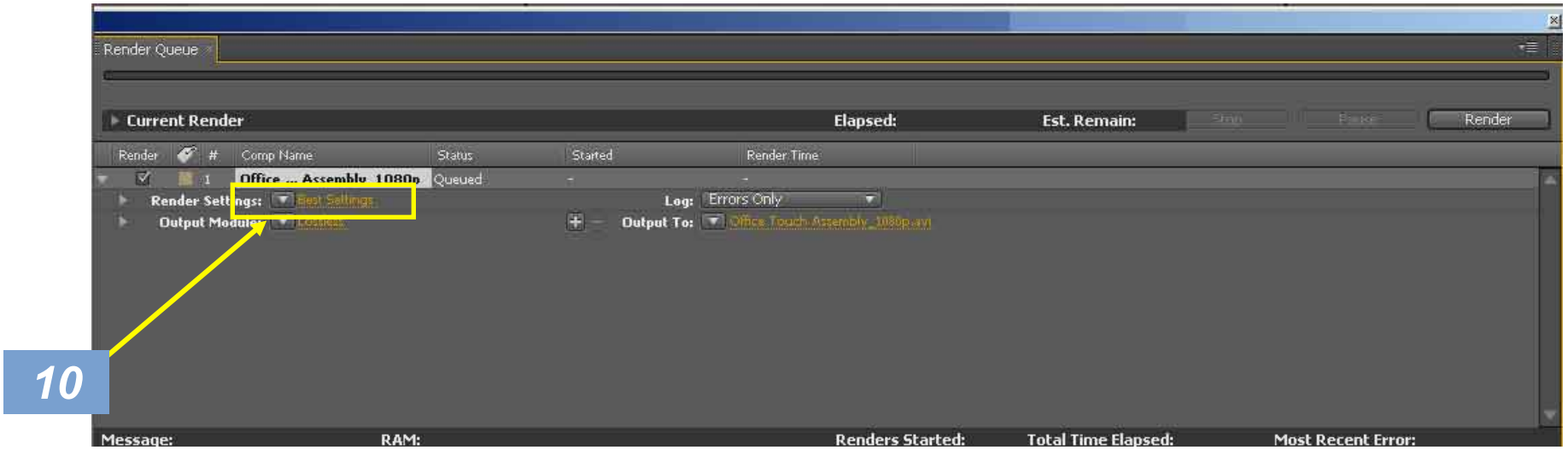
Custom Framing Vertical Video: After Effects CS4

- ***Step 9:*** Menu Bar: ***Select*** “Composition”>“Add to Render Queue”



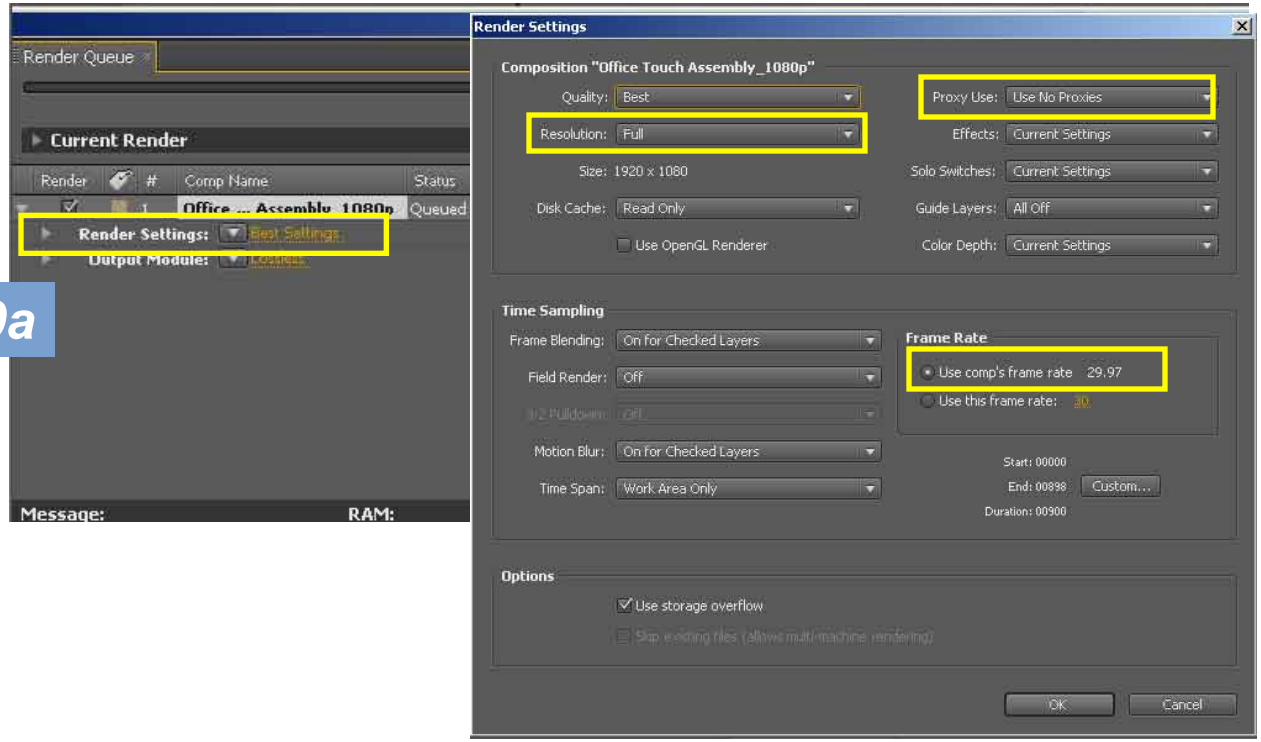
Custom Framing Vertical Video: After Effects CS4

- ***Step 10**: Launch* the Render Settings Module by clicking “Best Settings”



Custom Framing Vertical Video: After Effects CS4

- ***Step 10a*: Verify settings.**



Verify:

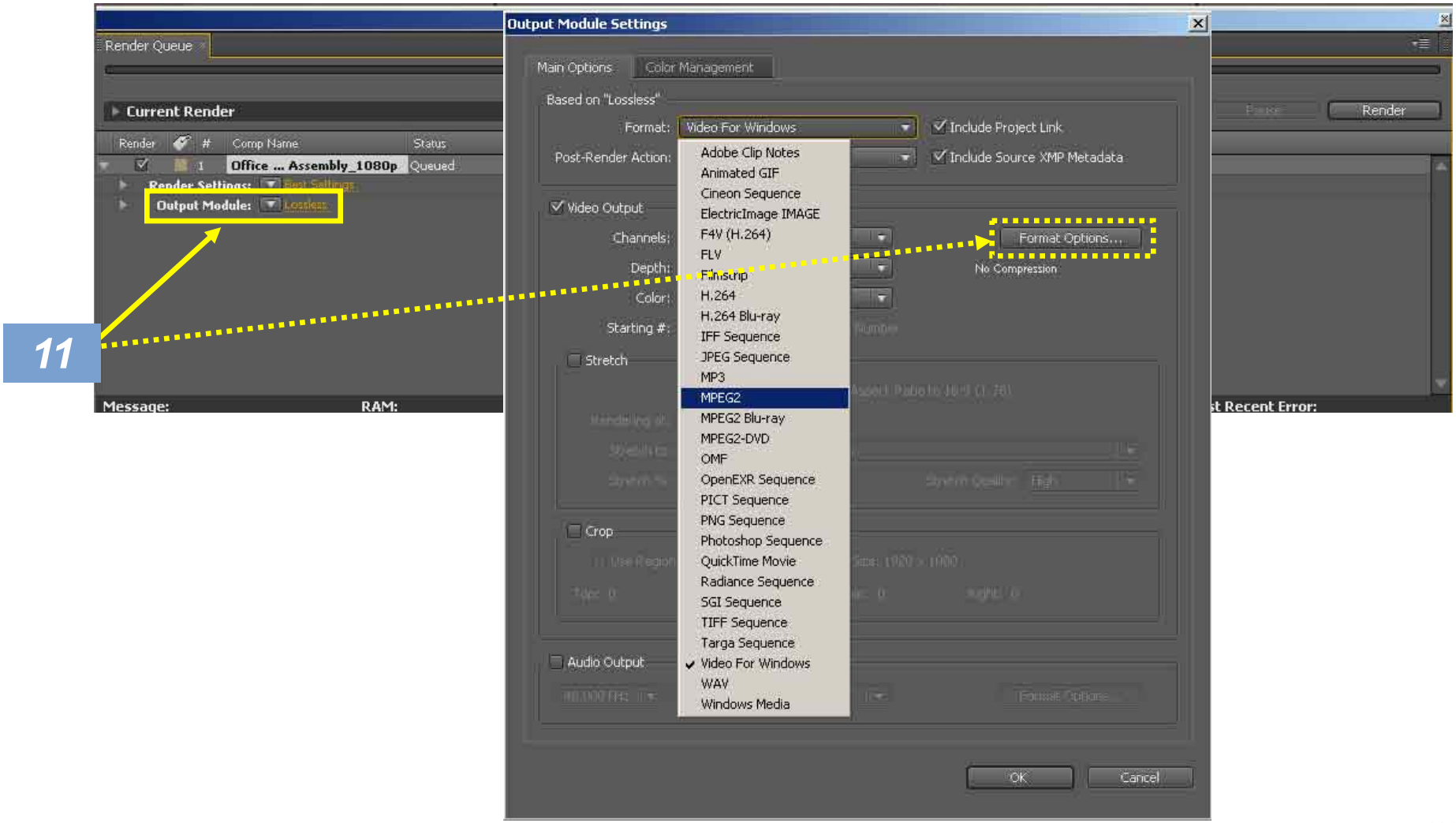
- ✓***Resolution = Full***
- ✓***Proxy Use = No Proxies***
- ✓***Use comp frame rate***

- ✓***Click "OK"***

Custom Framing Vertical Video: After Effects CS4

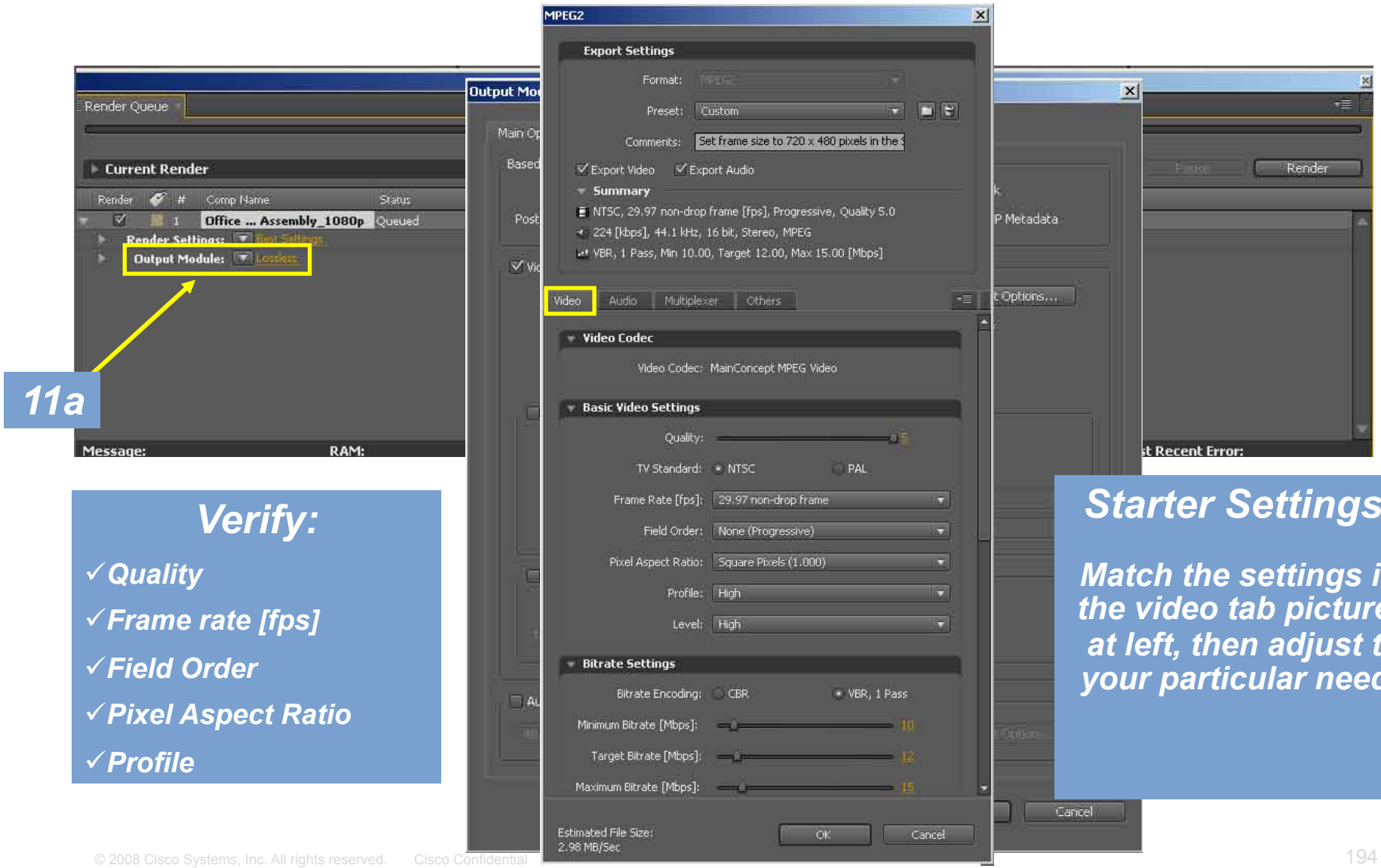
- ***Step 11*: *Launch*** the Output Module by clicking “Lossless”

  Select: Format > MPEG2. Then “Format Options...”



Custom Framing Vertical Video: After Effects CS4

- ***Step 11a*: *Select* :** The “Video” tab.



Verify:

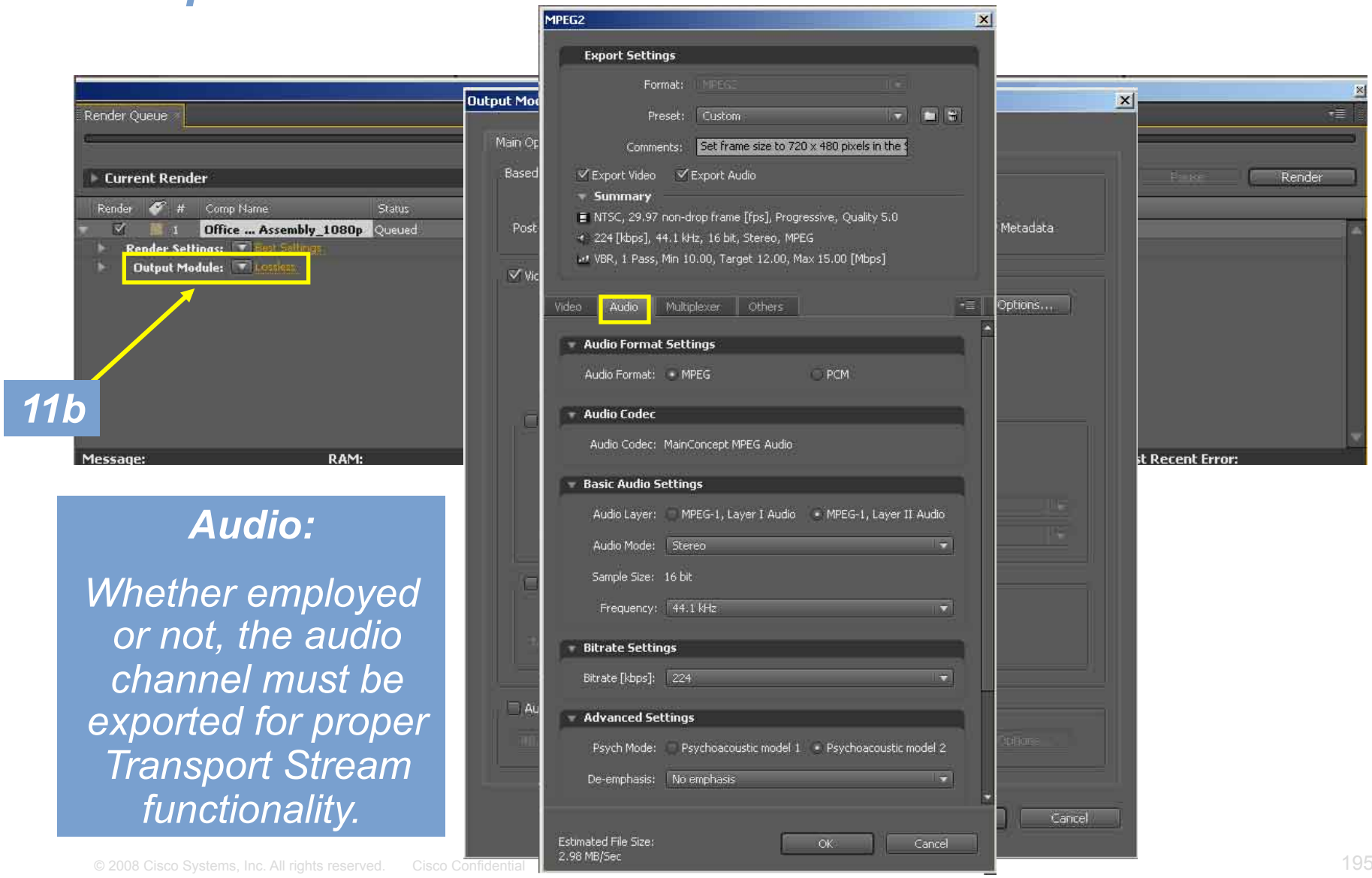
- ✓ *Quality*
- ✓ *Frame rate [fps]*
- ✓ *Field Order*
- ✓ *Pixel Aspect Ratio*
- ✓ *Profile*

Starter Settings:

Match the settings in the video tab pictured at left, then adjust to your particular needs

Custom Framing Vertical Video: After Effects CS4

- ***Step 11b*: *Select*** : The “Audio” tab.

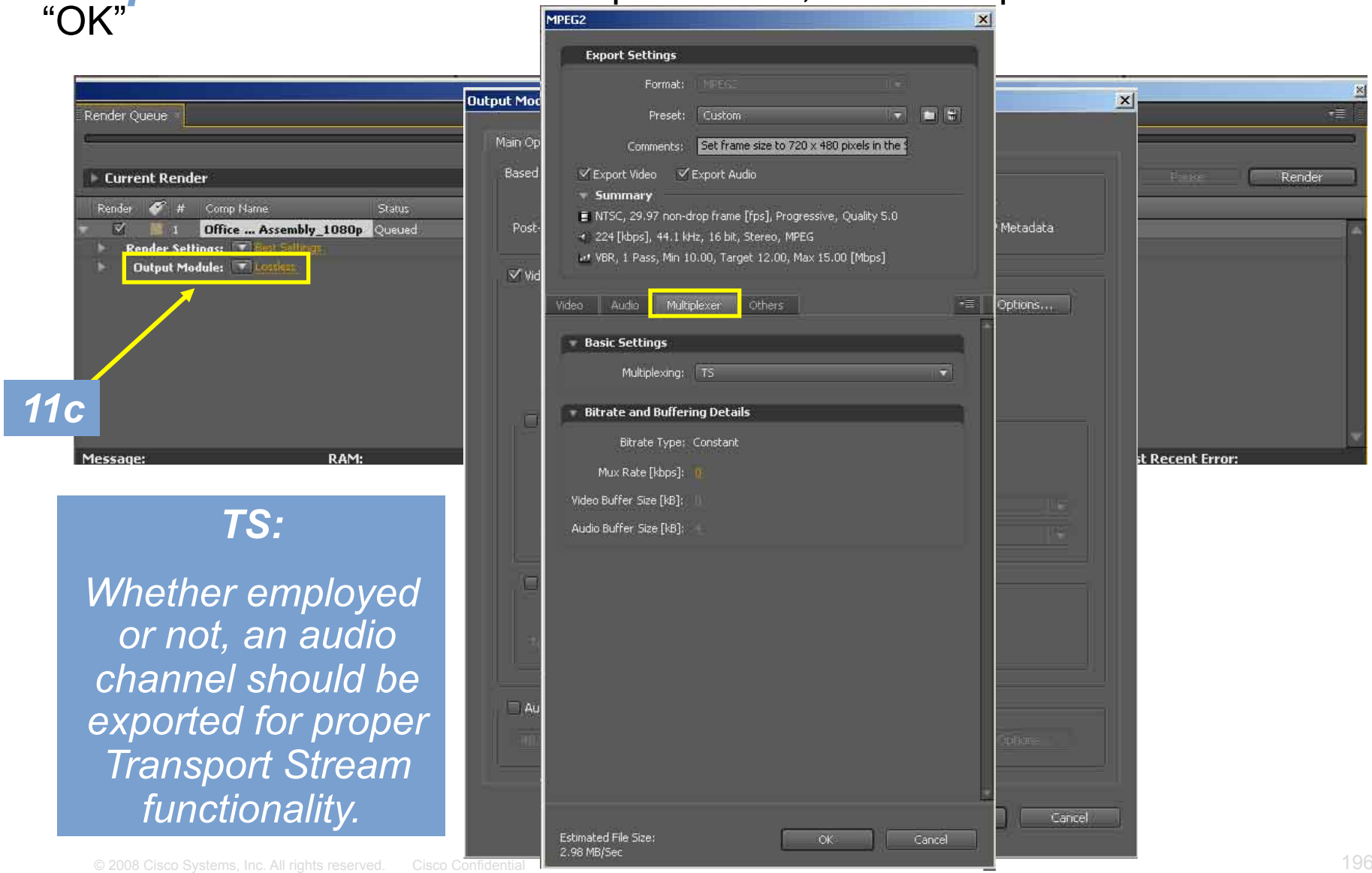


Audio:

Whether employed or not, the audio channel must be exported for proper Transport Stream functionality.

Custom Framing Vertical Video: After Effects CS4

- ***Step 11c*: *Select*** : The “Multiplexer” tab, and “Multiplexer =TS”. “OK”

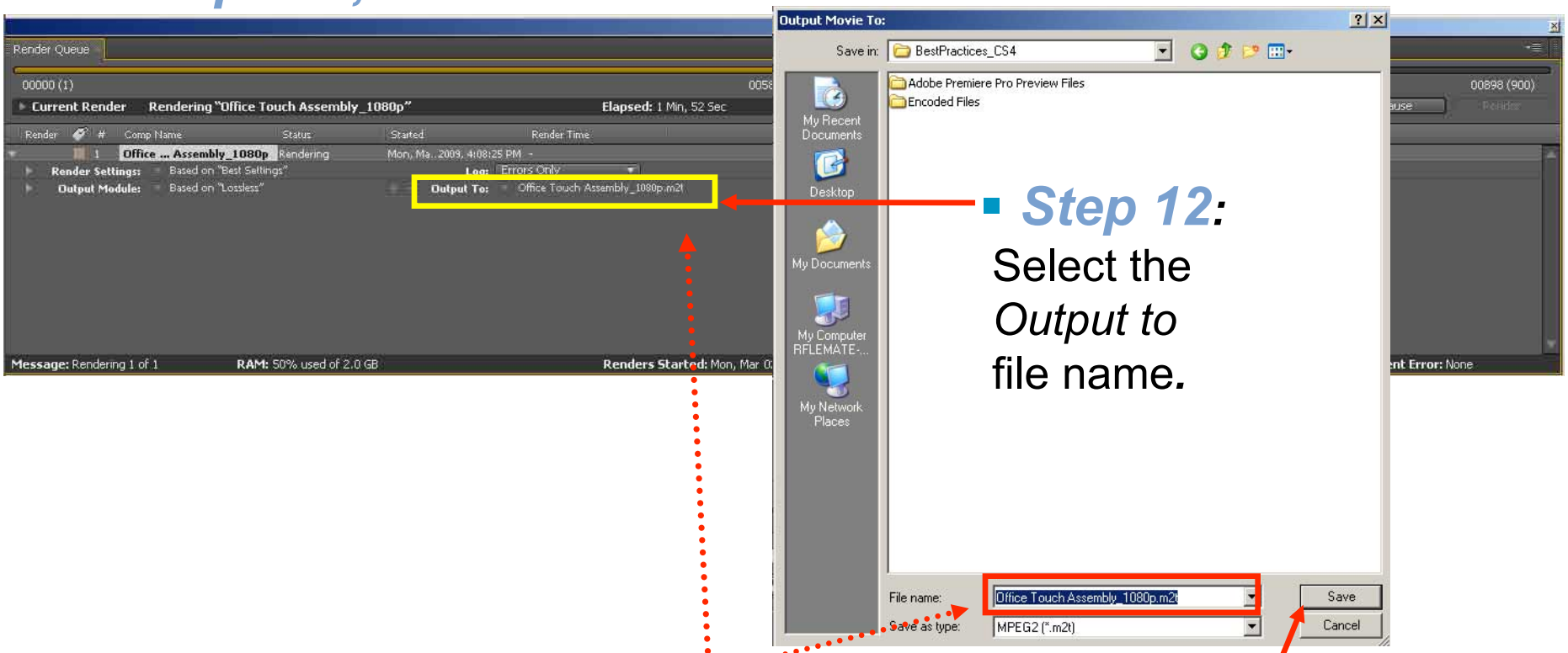


TS:

Whether employed or not, an audio channel should be exported for proper Transport Stream functionality.

Custom Framing Vertical Video: After Effects CS4

- ***Steps 12, 13:***



- ***Step 12:*** Select the *Output to* file name.

- ***Step 13:*** Click, ***Save.***

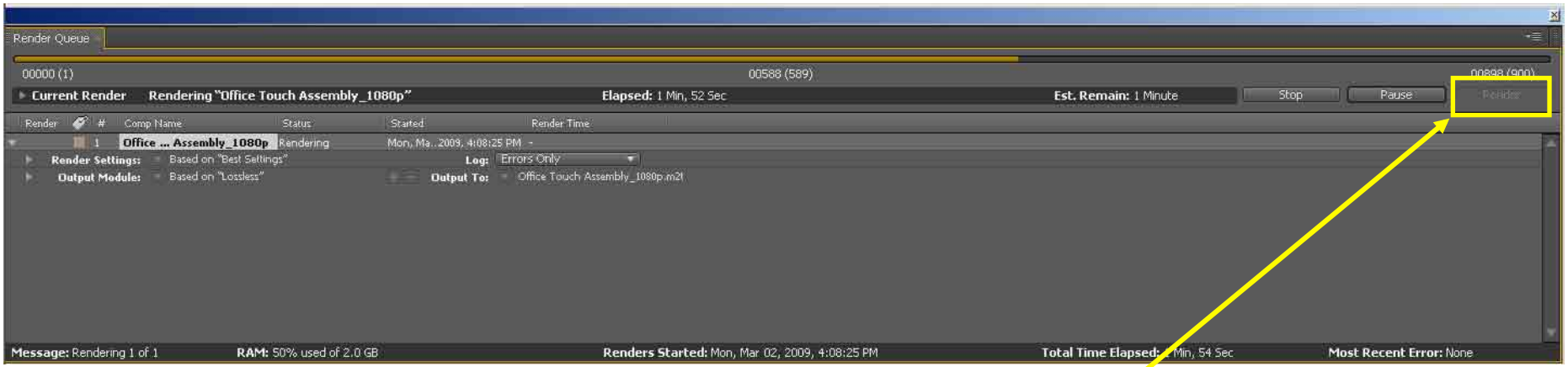
Note: This "Save" means "Save file to" and does not start the rendering

Notice *the file type: .m2t = MPEG2-TS*

Change to ".mpg" for playback on DMP.

Custom Framing Vertical Video: After Effects CS4

- ***Steps 14: Generating your MPEG2-TS file.***



- ***Step 14:***
  Click, ***Render.***
  to output the MPEG2-TS file.

- ***(optional) :*** To save on file size you may also use the VLC player to transcode the "mp2v" codec to an h.264 codec.

  See "MPEG2 in VLC"

Custom Framing Vertical Video: After Effects CS4

- ***Step 15*: File extension change.**

.m2t to .mpg

Change the extension from ".m2t" to ".mpg:.

Play with your preferred player.

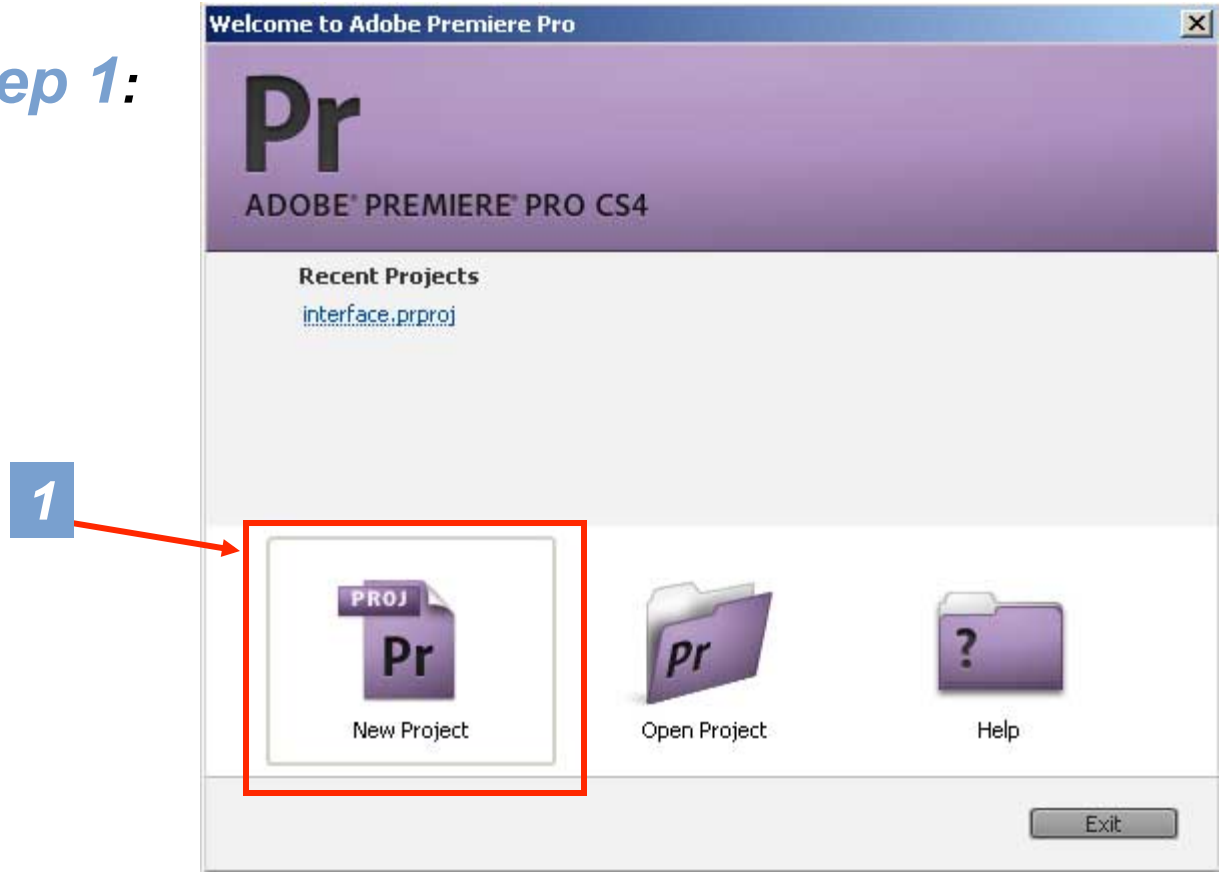
NOTE: Currently, the Quicktime MPEG add-on will not play MPEG2-TS files.

Custom Framing Vertical Video: Premiere Pro CS4

- Vertical and non-standard framed videos should be set up at the creation of a Premiere project.
- Vertical and non-standard framed videos should be saved in an uncompressed, cross-platform AVI format and used as a source to generate a custom MPEG-2 TS file in an appropriate video editing or compositing software package.
- Exporting via Adobe Media Encoder may introduce letter or pillar boxing to the final video.

Custom Framing Vertical Video: Premiere Pro CS4

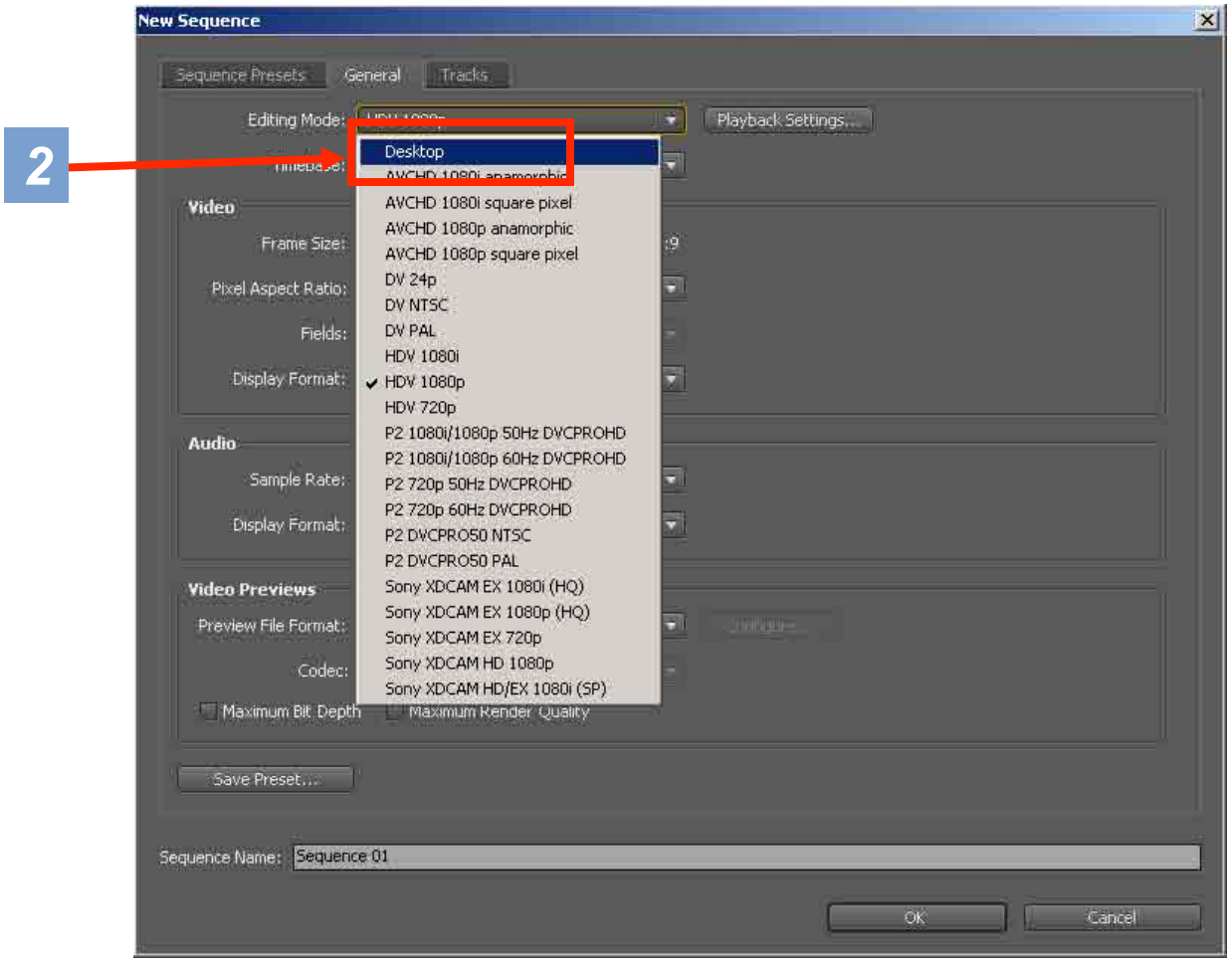
- *Step 1:*



- *Note: These steps assume a non-interlaced, square pixel, output playing within a template. Where necessary, convert your source footage to a progressive, square pixel state at a maximum screen size of 1366x768.*

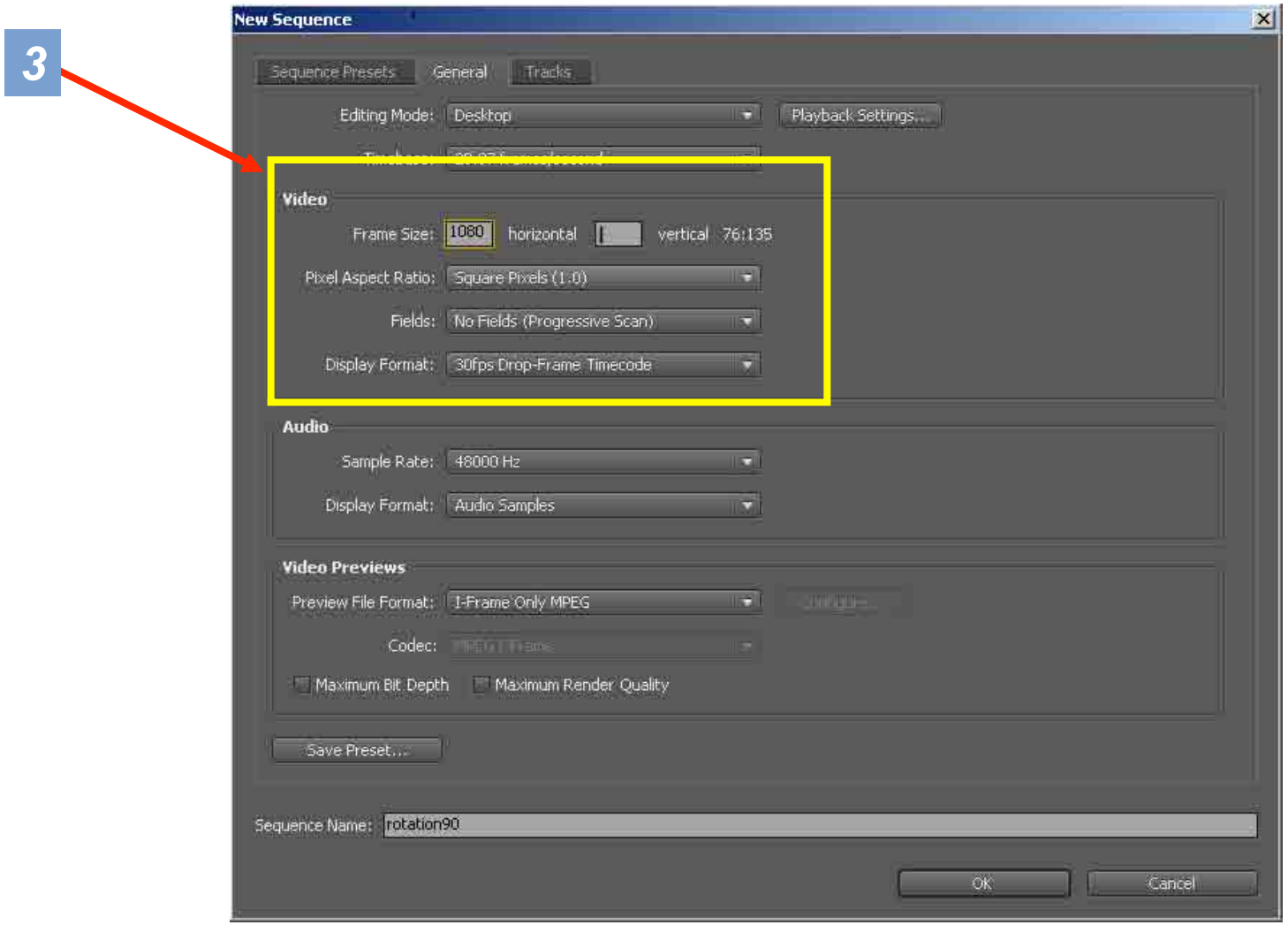
Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 2*: Select "Desktop" for the "Editing Mode".**



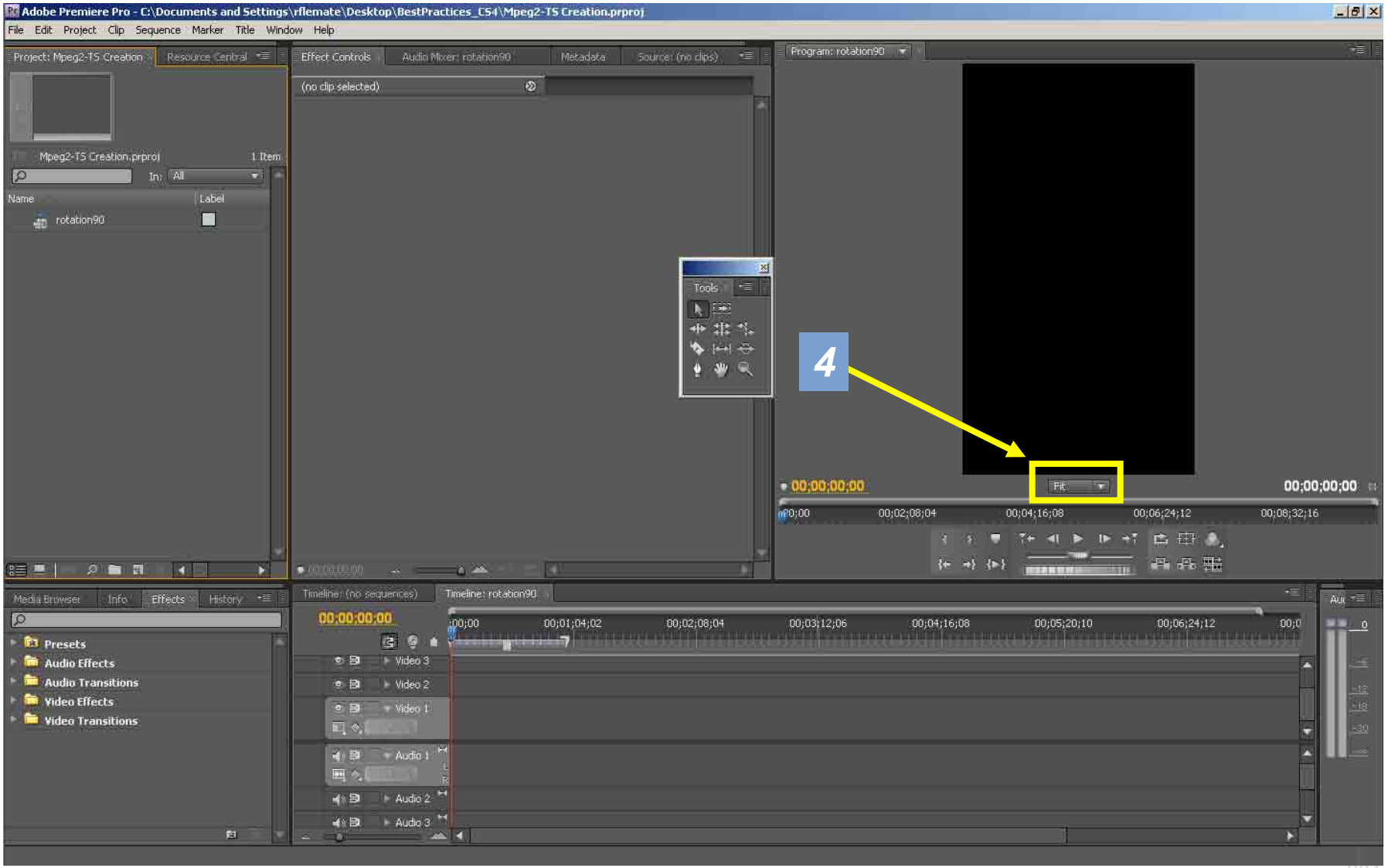
Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 3*: Enter a desired frame size and match the other settings.**



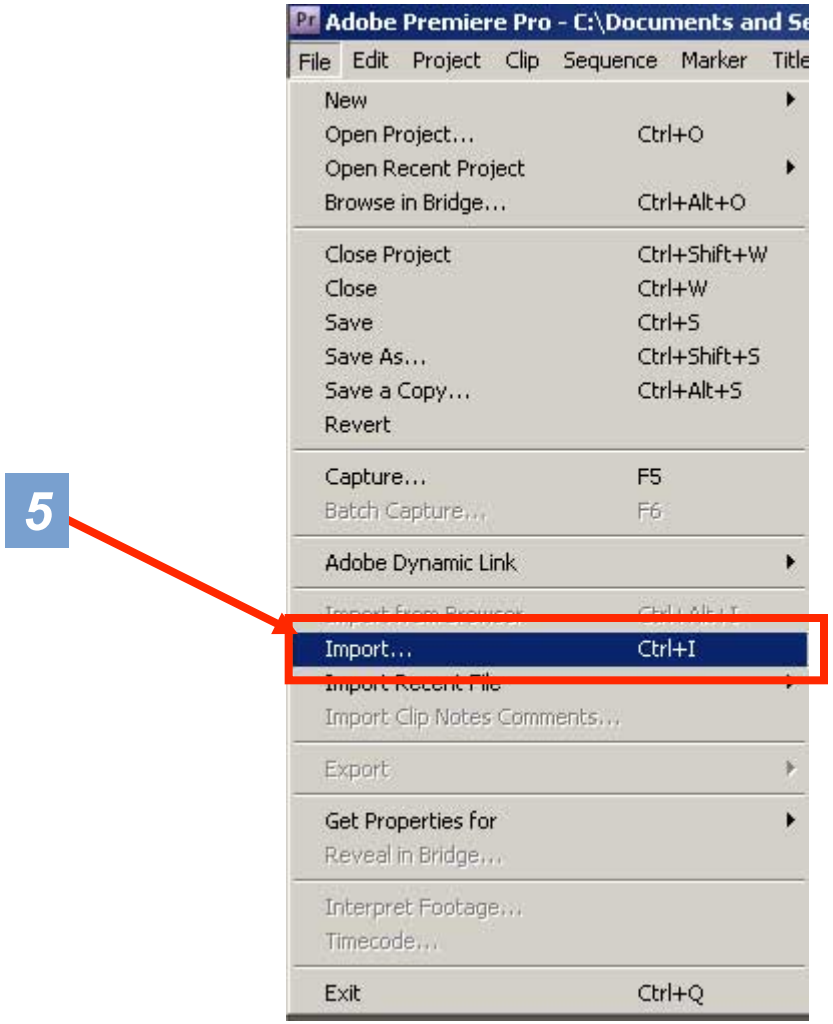
Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 4*: You may need to fit your vertical frame into view.**



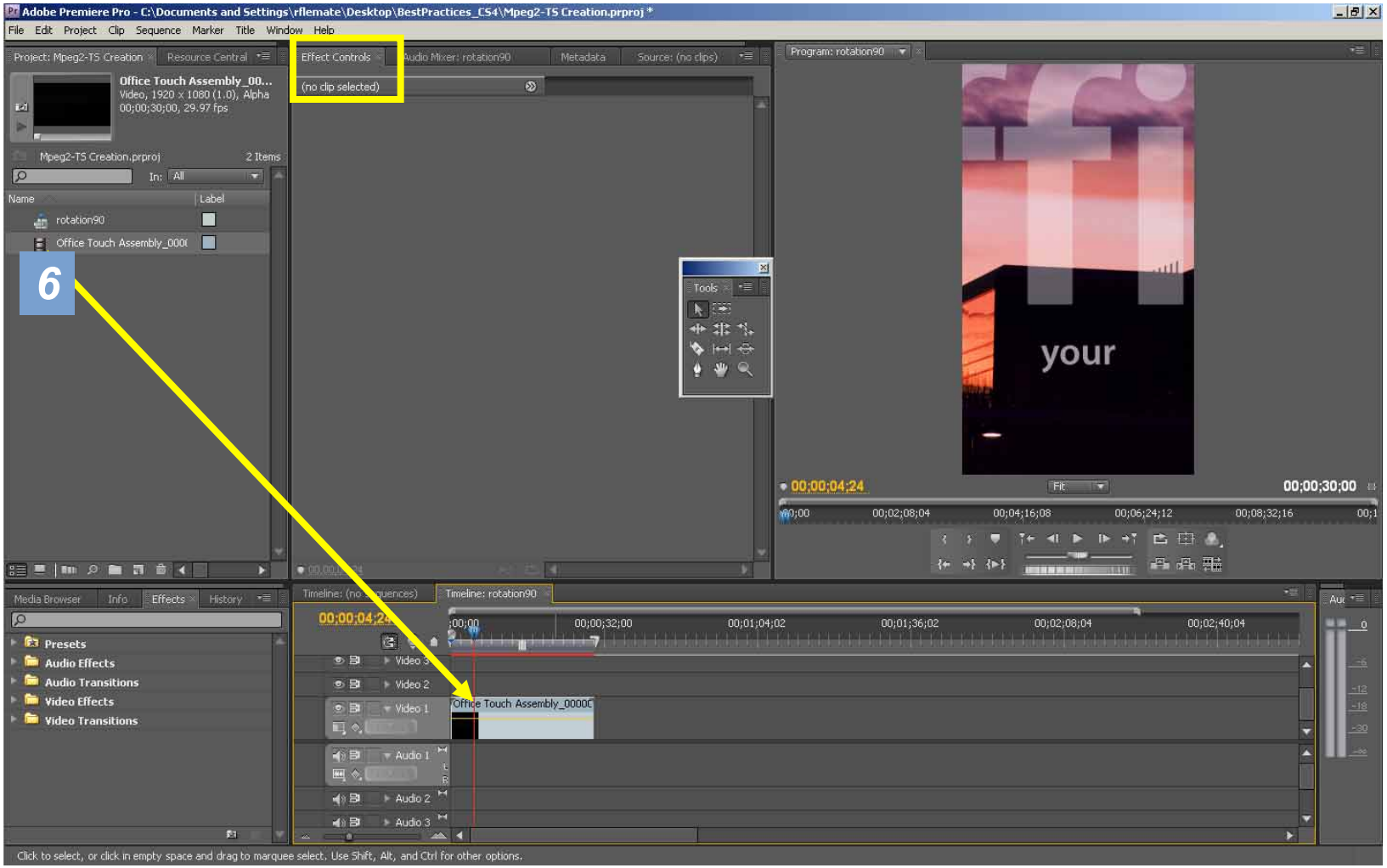
Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 5*: Import your footage or image sequence.**



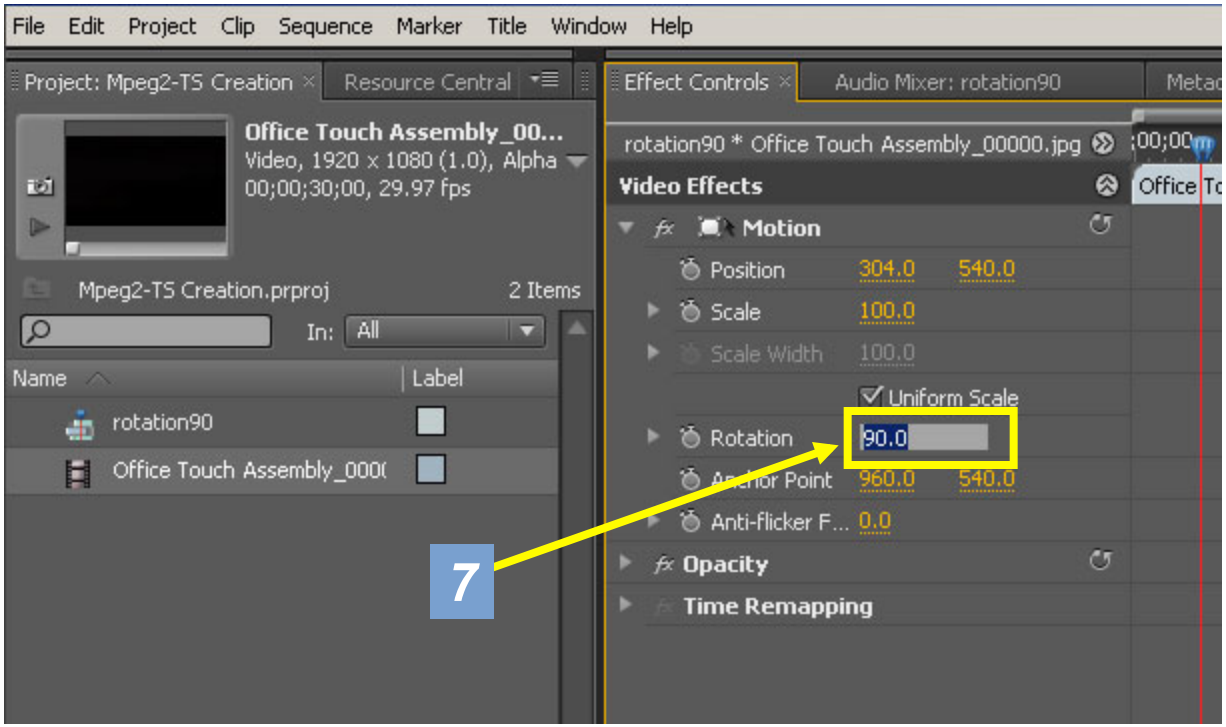
Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 6*: Add footage to timeline then change footage orientation using the "Effects Control" tab.**



Custom Framing Vertical Video: Premiere Pro CS4

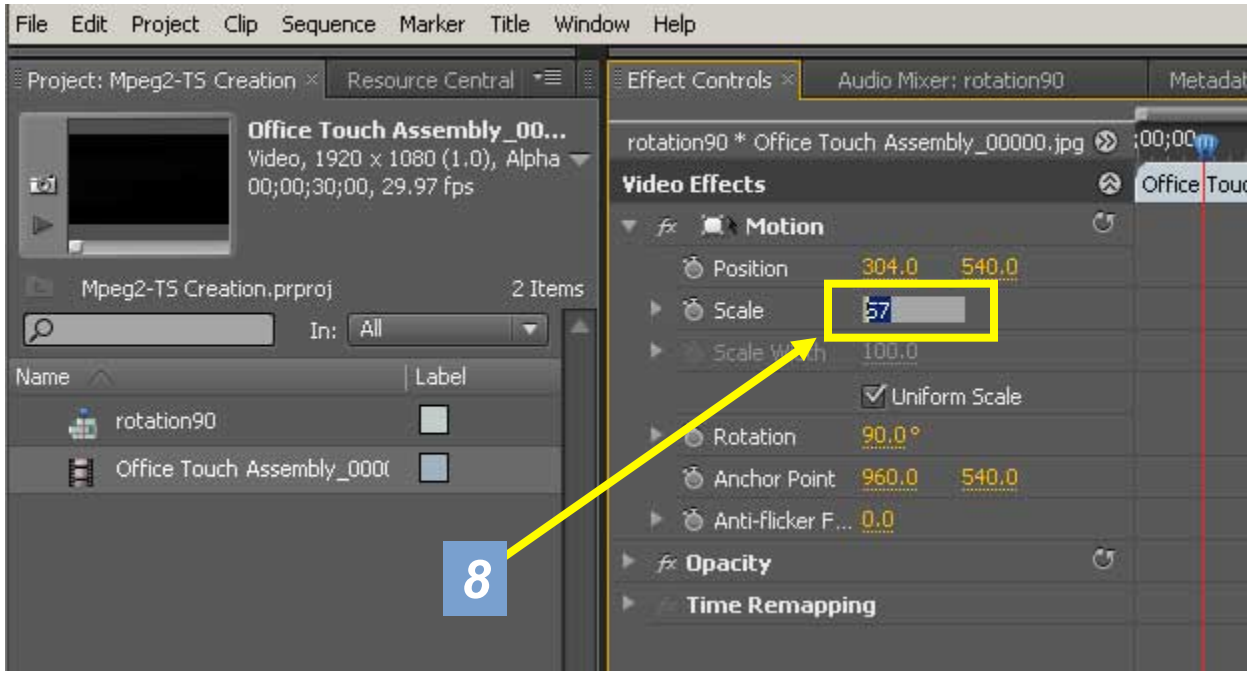
- ***Step 7*: Select "Effect Controls>Motion>Rotation=90 or 270.**



- ***Step 8*: Export rotated video to an appropriate file format.**

Custom Framing Vertical Video: Premiere Pro CS4

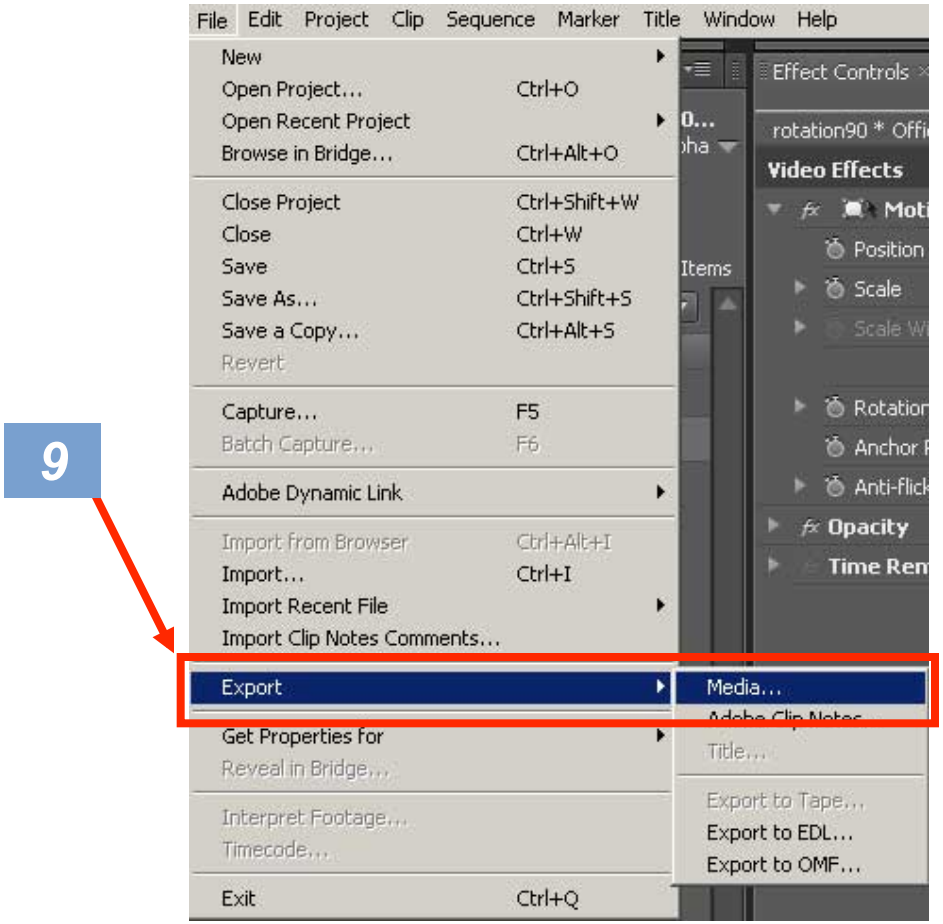
- ***Step 8*: Select "Effect Controls>Motion>Scale=Enter a value.**



- ***Step 10*: Export rotated video to an appropriate file format.**

Custom Framing Vertical Video: Premiere Pro CS4

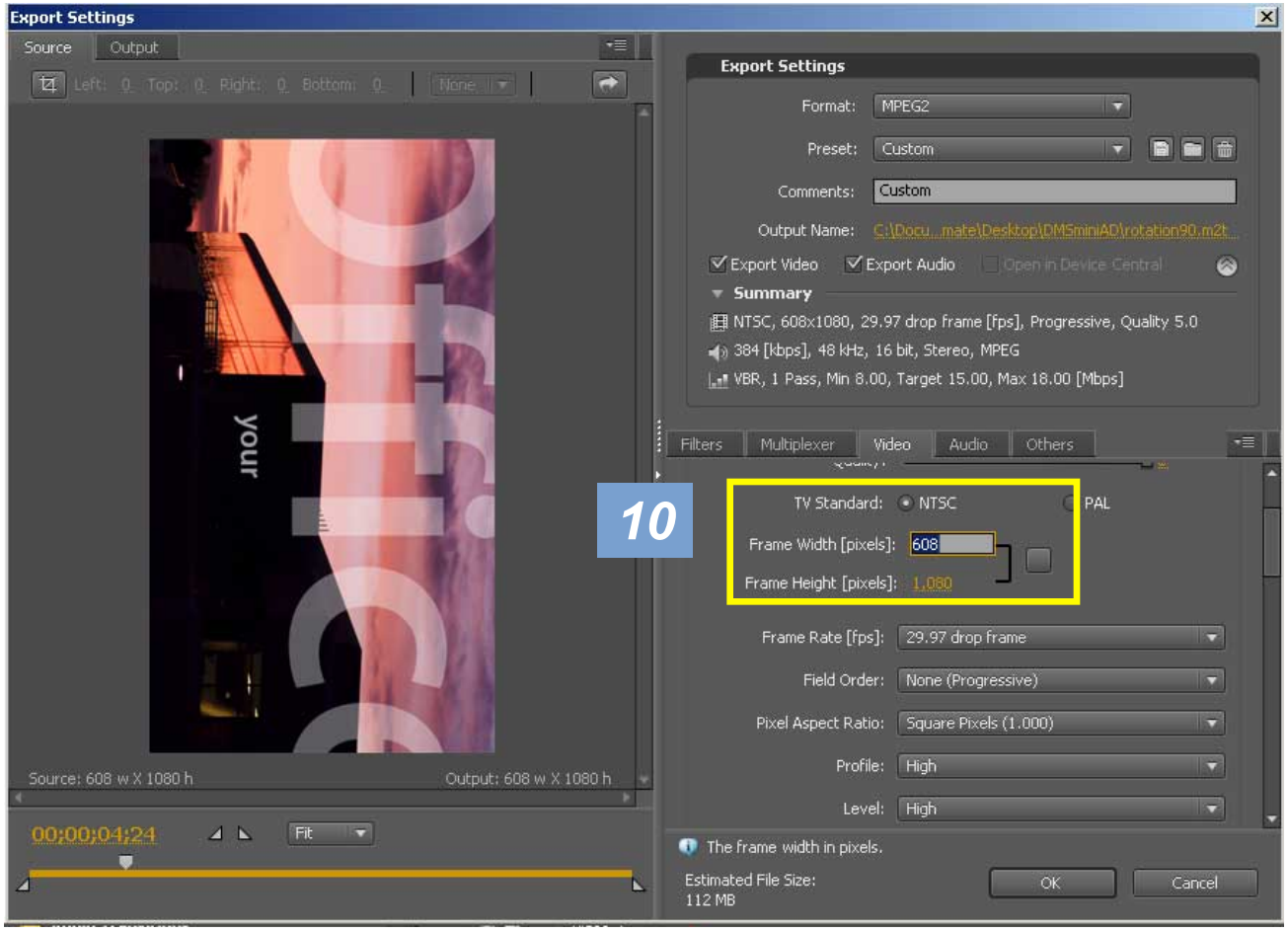
- ***Step 9*: Select File>Export>Media...**



This Export rotated video to an appropriate file format.

Custom Framing Vertical Video: Premiere Pro CS4

- *Step 10*: ***Export>Media***

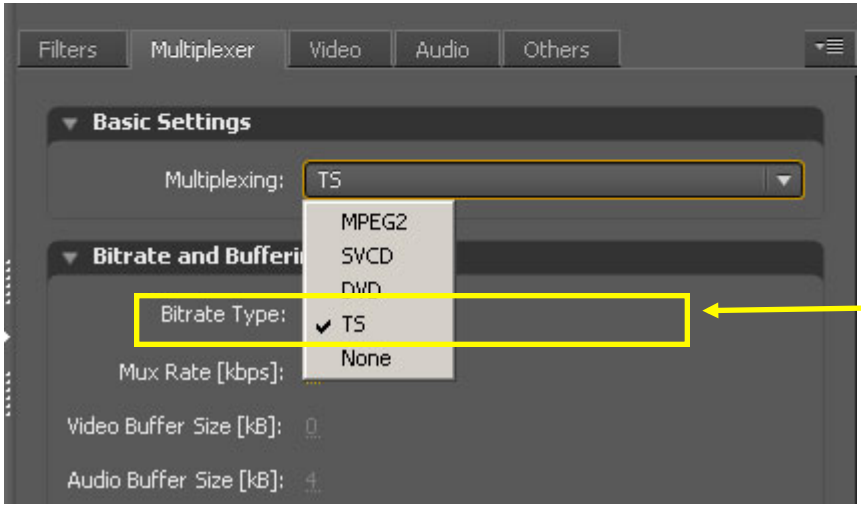


NOTE: Uncheck the "Constraint" box and be sure to enter you desired vertical format parameters.

Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 11*: *The Multiplex, Video, and Audio tabs:***

Multiplex 11

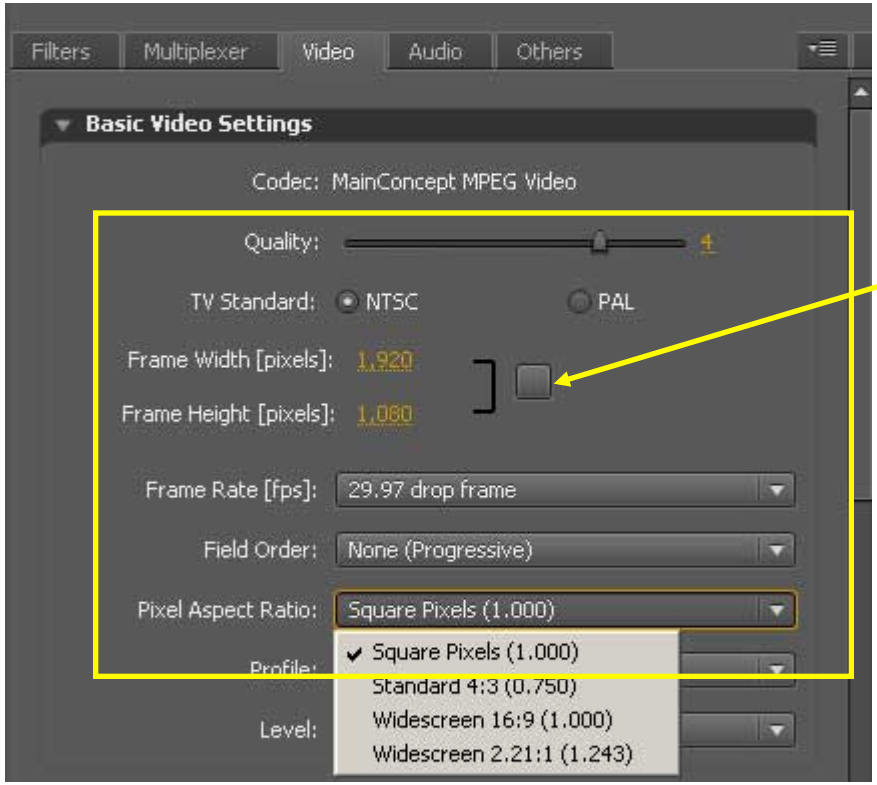


11

Select the "TS" (transport stream) setting. In CS4, bitrates are now handled in the Video tab.

Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 11a*: *The video tab.***



Video 11a

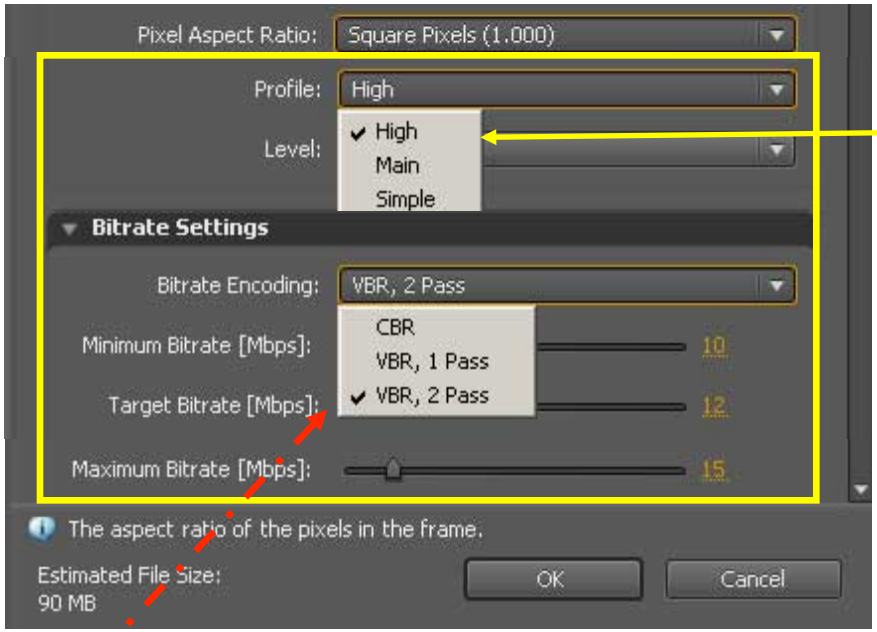
11a

These are the general video settings.

Be sure to uncheck the constraint and match your Premiere project's custom width and height.

Custom Framing Vertical Video: Premiere Pro CS4

- ***Steps 11b & 11c*: *The video tab.***



Video 11b, 11c

Although not a bitrate par setting, selecting the proper profile impacts the range of bitrates available. For 720p video output and greater, choose the "High" profile for DMP playback.

VBR, 1 Pass is a good compressor, "2 Pass" will create a smaller file size and doubles rendering time.

A good bitrate range is between 8MB/s to 18 MB/s.

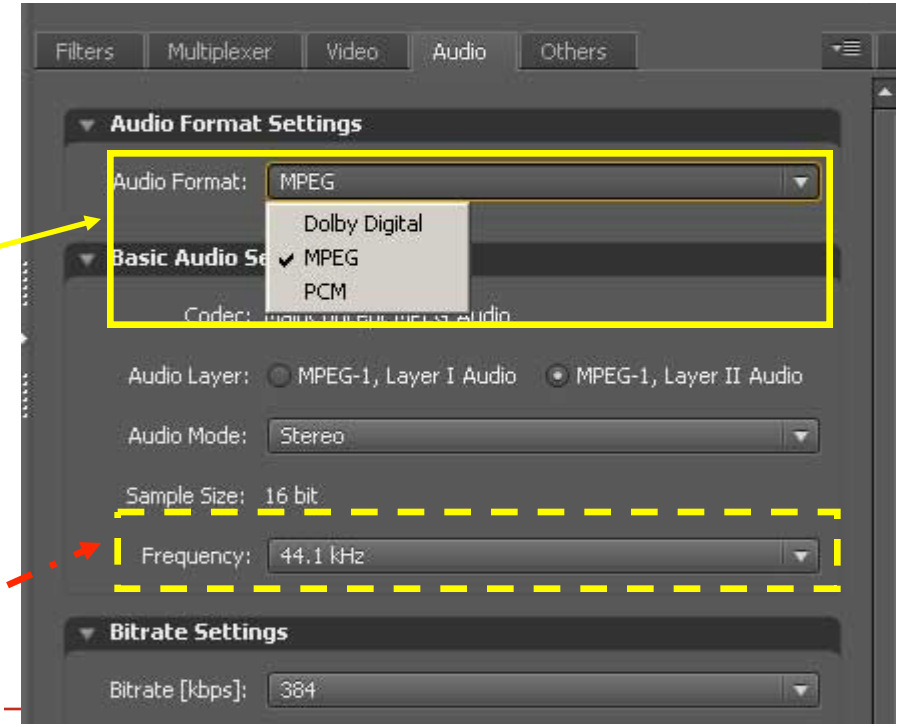
As with any data limiter, test your results for visual quality.

Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 12*: Audio tab: Select MPEG format**

Audio tab: 12

Whether audio is used or not, include audio output to ensure proper functionality to the transport stream (TS).



Audio Frequency Guide:

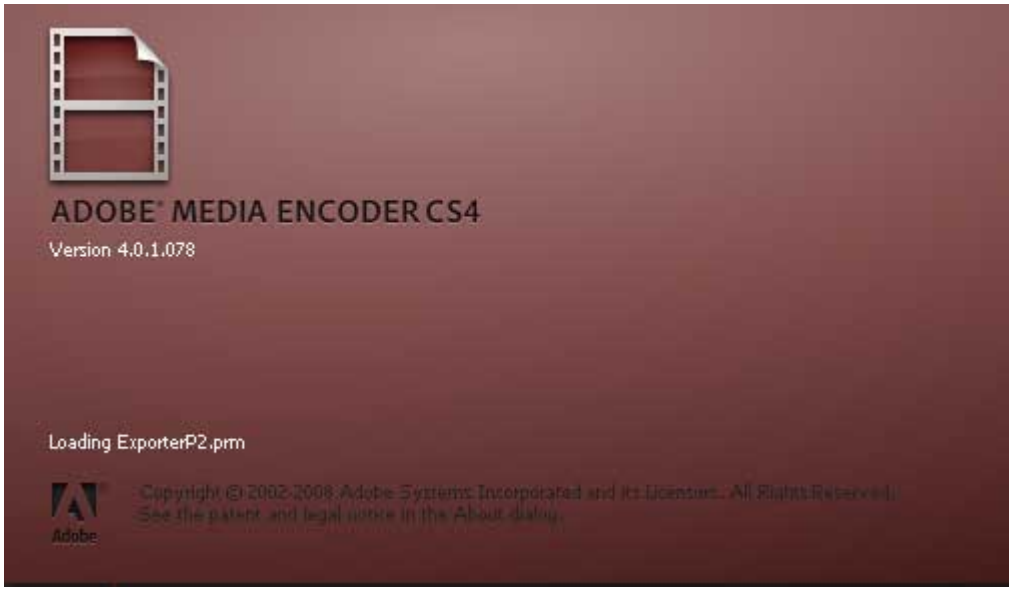
48.0 kHz: DVD quality.

44.1 kHz: CD quality.

32.0 kHz: Generic

Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 13*: Clicking "OK" will launch Adobe Media Encoder.**



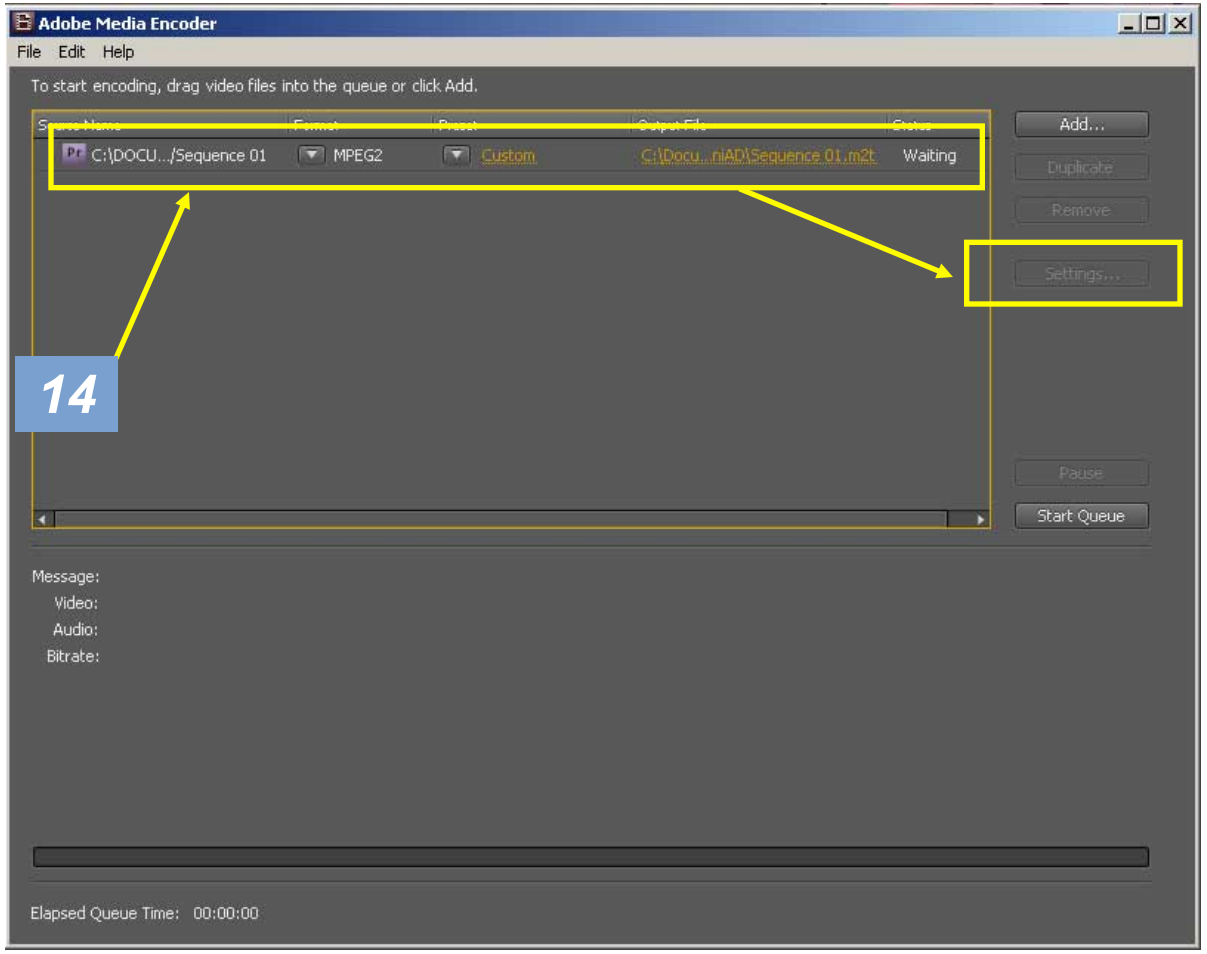
Remember your export settings:

Since Adobe Media Encoder is an independent application, verify all parameters match those entered into the Premiere Pro Export.

Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 14*: Adobe Media Encoder: Matching parameters.**

Highlight your project in the queue and click the "Settings" button.



Since Adobe Media Encoder is an independent application, here you will double -check all parameters.

Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 15*: Adobe Media Encoder: Confirm settings.**

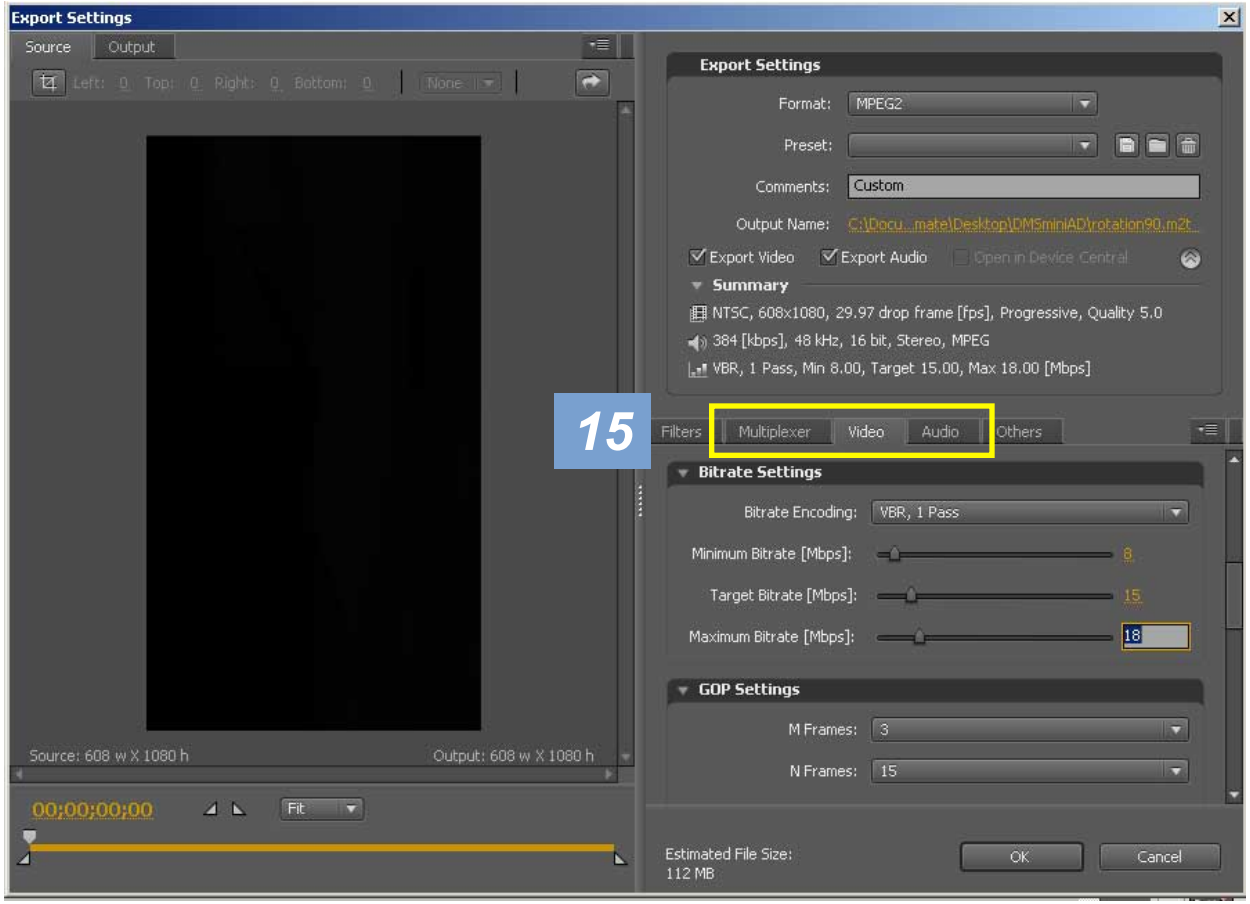
Video

Confirm all parameters and that the AME tabs match the Premiere Export settings for:

"Multiplexer"

"Video"

"Audio"



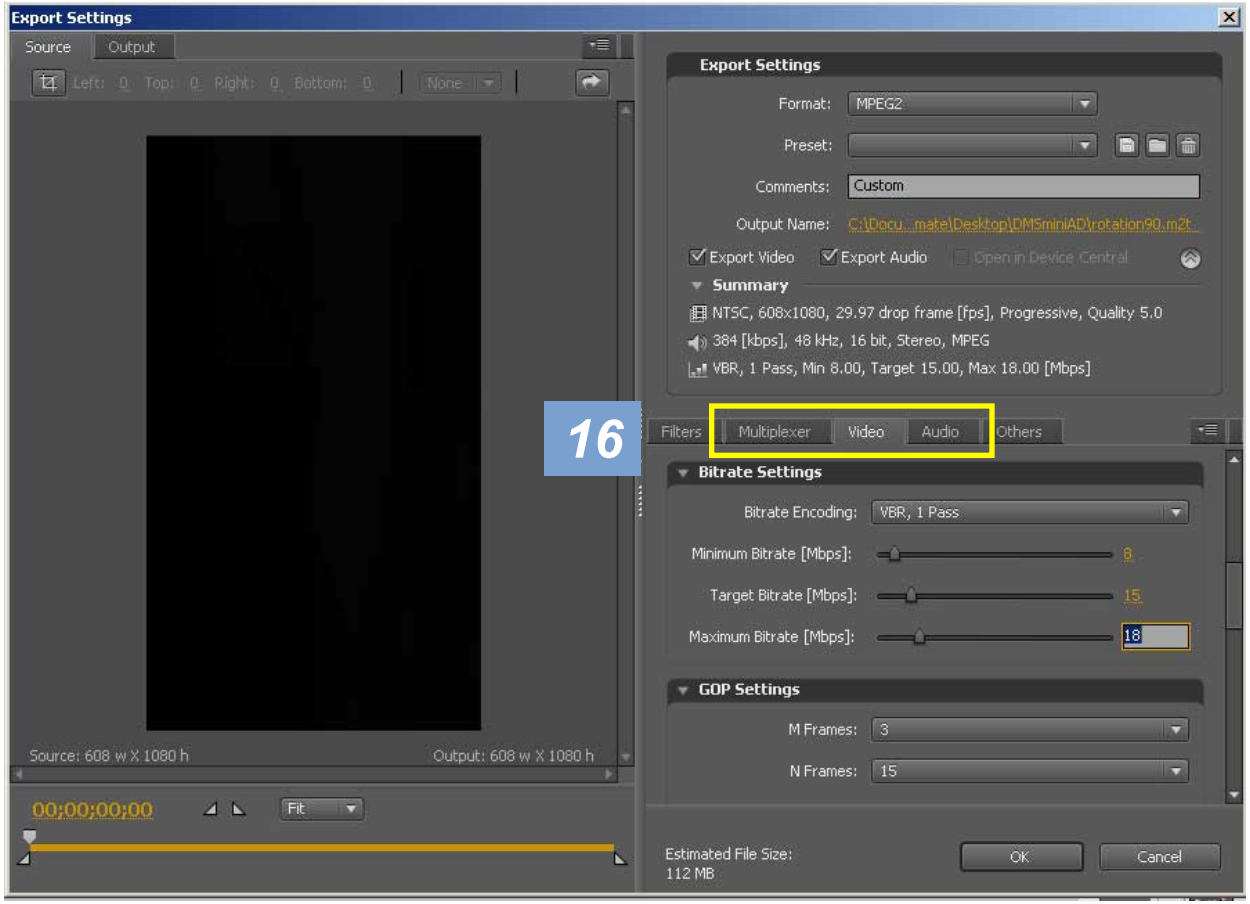
Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 16*: Adobe Media Encoder: Include audio export.**

Audio

Include an audio export for proper transport stream function even if the audio track is empty.

Click "***OK***"

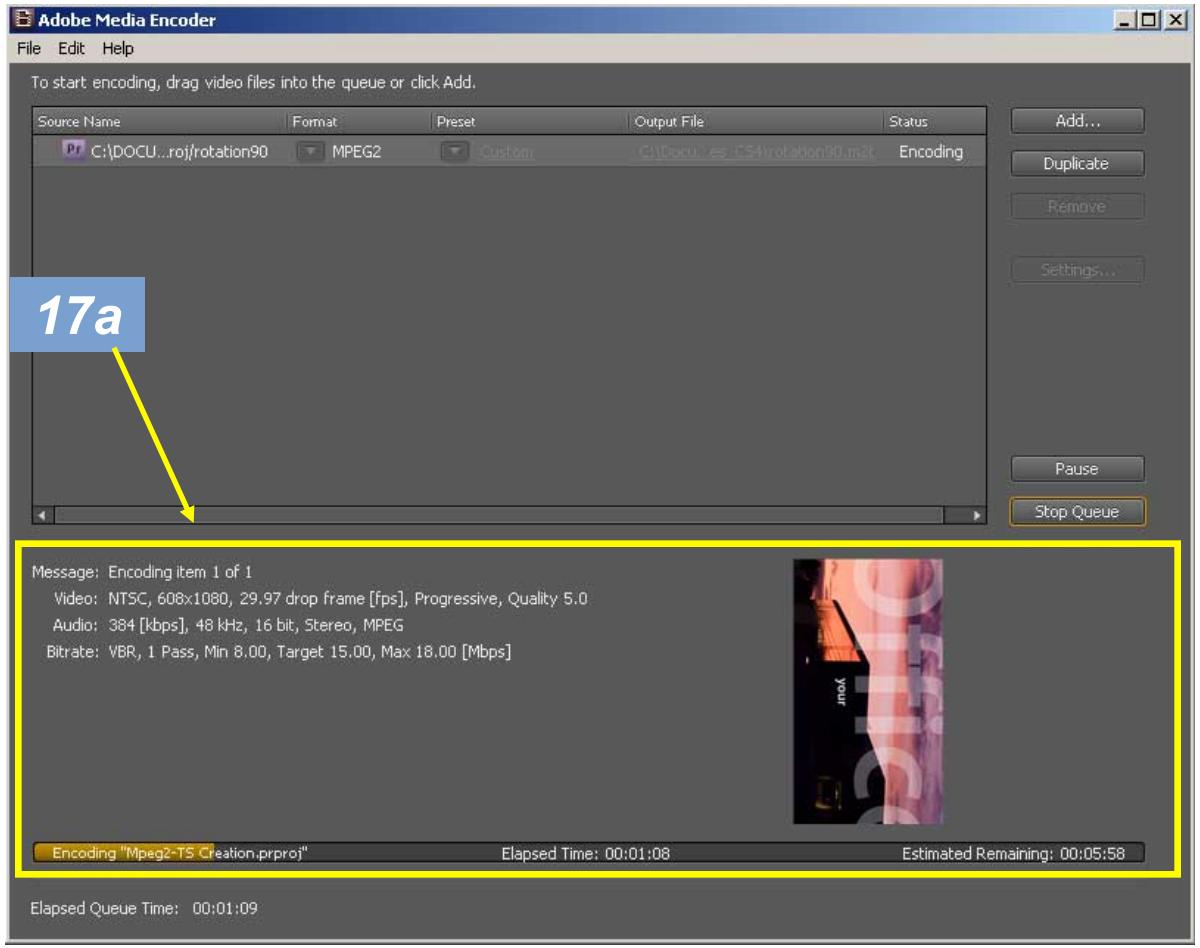


Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 17, 17a*: Adobe Media Encoder: Click "Start Queue"**

Started Queue

Once the queue is started, you should see the progress bar and a small render window which should reflect the vertical format.



Custom Framing Vertical Video: Premiere Pro CS4

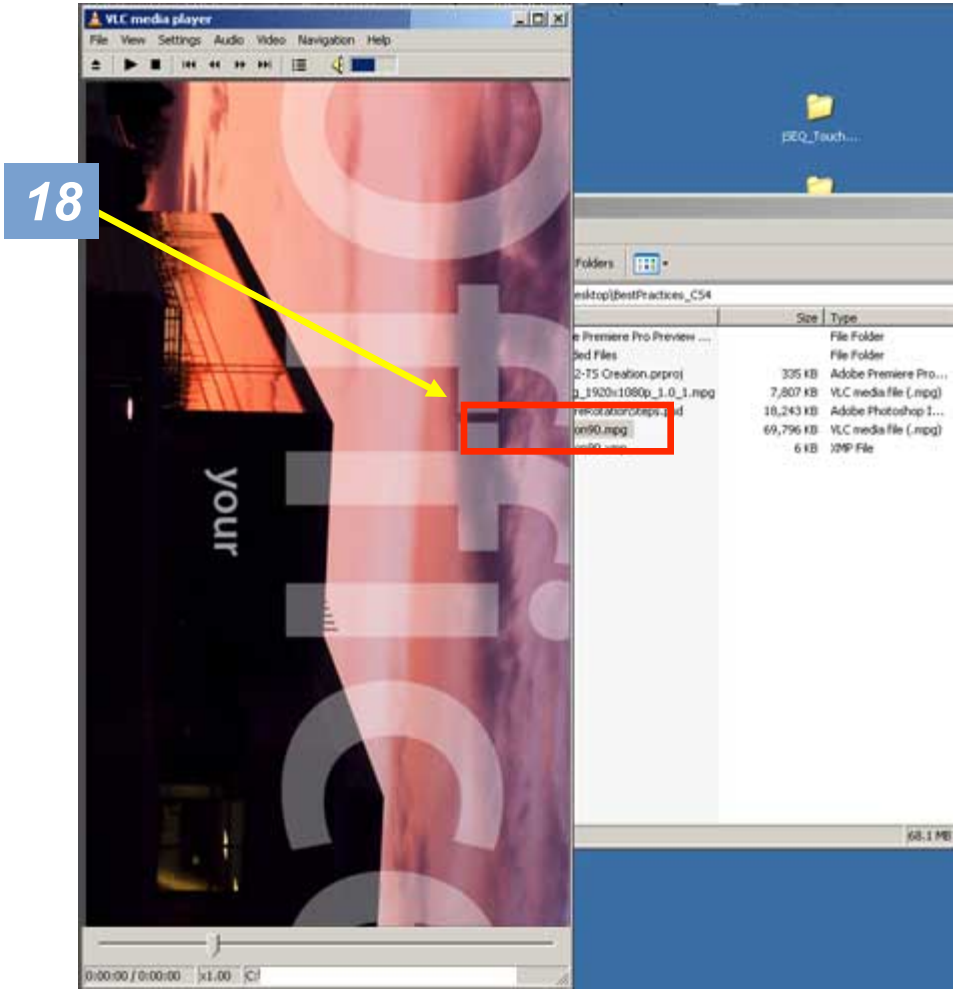
- ***Step 18*: File extension change.**

.m2t to .mpg

Change the extension from ".m2t" to ".mpg"

Play with your preferred player.

NOTE: Currently, the Quicktime MPEG add-on will not play MPEG2-TS files.



Custom Framing Vertical Video: Premiere Pro CS4

- ***Step 19*: File extension change.**

.m2t to .mpg

Change the extension from ".m2t" to ".mpg:.

Play with your preferred player.

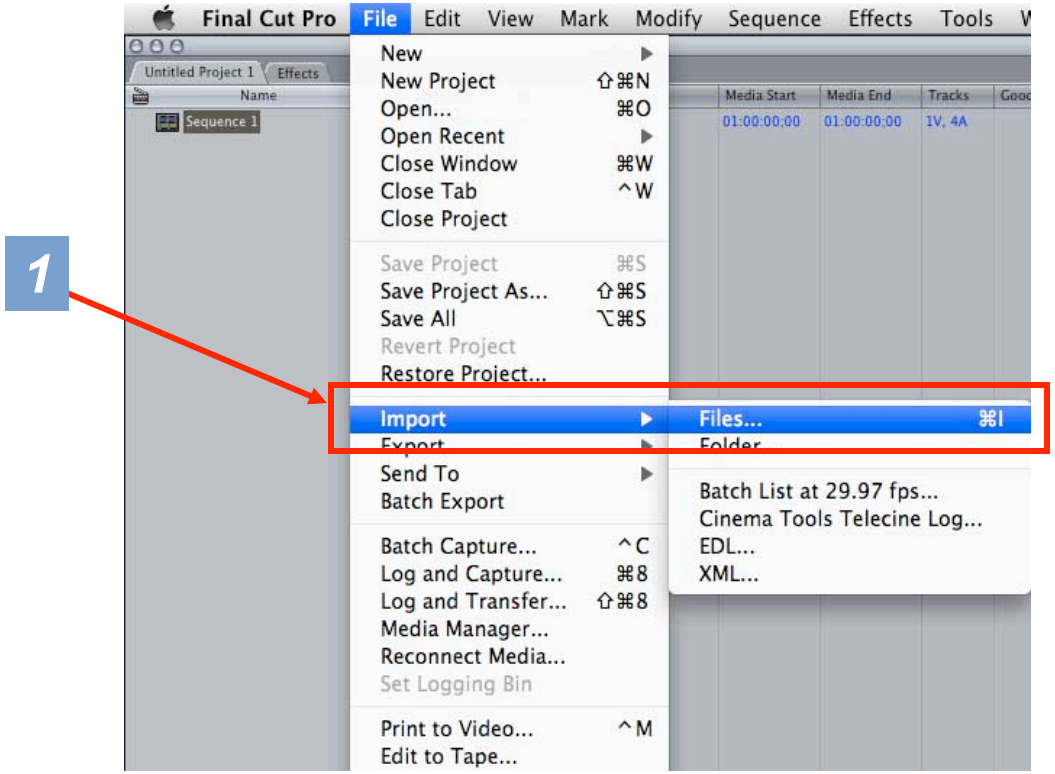
NOTE: Currently, the Quicktime MPEG add-on will not play MPEG2-TS files.

Custom Framing Vertical Video: in FCP

- Final Cut Pro video may be exported via Compressor to an MPEG-2 TS, square pixel, format which uses either the 4:3 or 16:9 frame aspect ratio.
- Vertical and non-standard framed video (3:4 or 9:16) should be saved in an uncompressed, cross-platform AVI format and used as the source to generate a custom MPEG-2 TS file using an appropriate video editing or compositing software package.
- Though the VLC media player is available for the Apple platform, it is recommended that VLC formatting and transcoding be performed on a PC to minimize the impact between software platform differences.

Custom Framing Vertical Video: in FCP

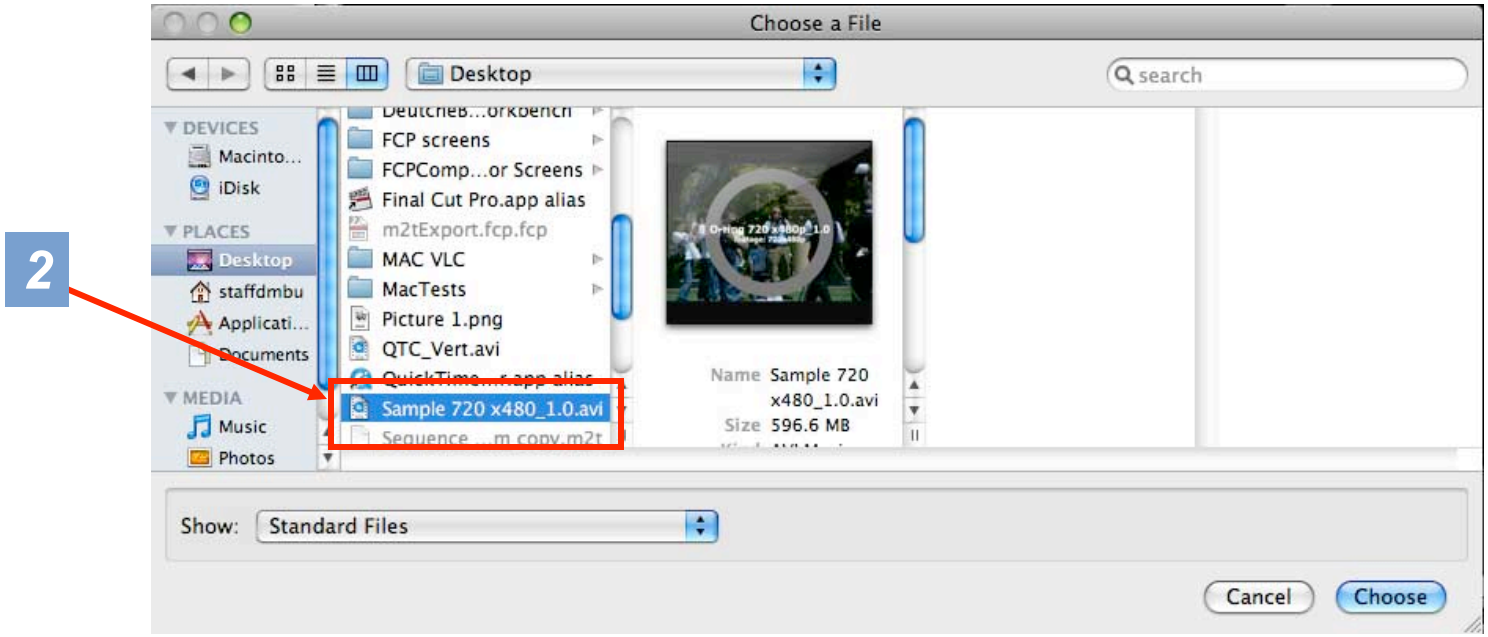
- ***Step 1*: File>Import> Files**



- ***Note*: These steps assume a non-interlaced, square pixel, output playing within a template. Where necessary, convert your source footage to a progressive, square pixel state at a maximum screen size of 1366x768.**

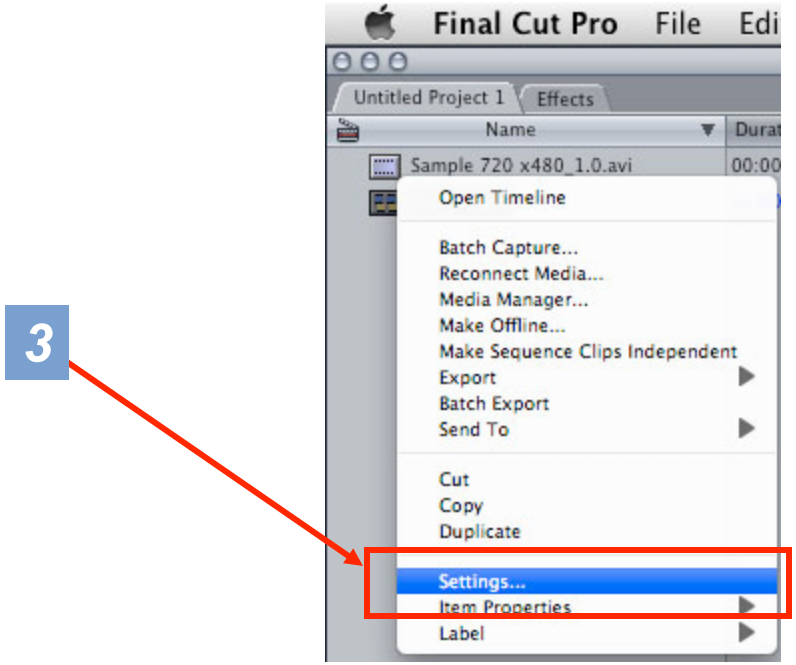
Custom Framing Vertical Video: in FCP

- ***Step 2*: Select your source file(s) or Export project**



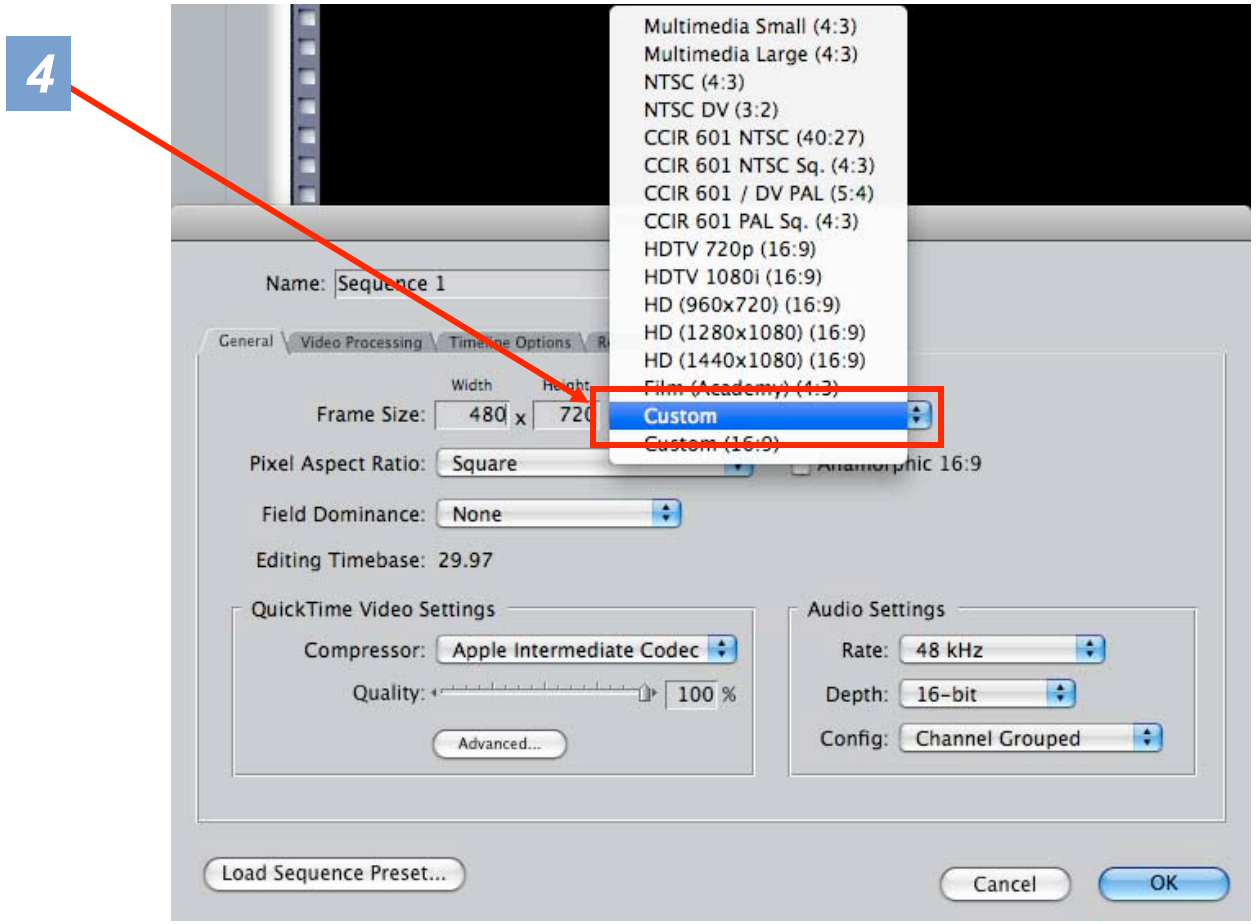
Custom Framing Vertical Video: in FCP

- ***Step 3*: Select the Sequence Settings to adjust framing.**



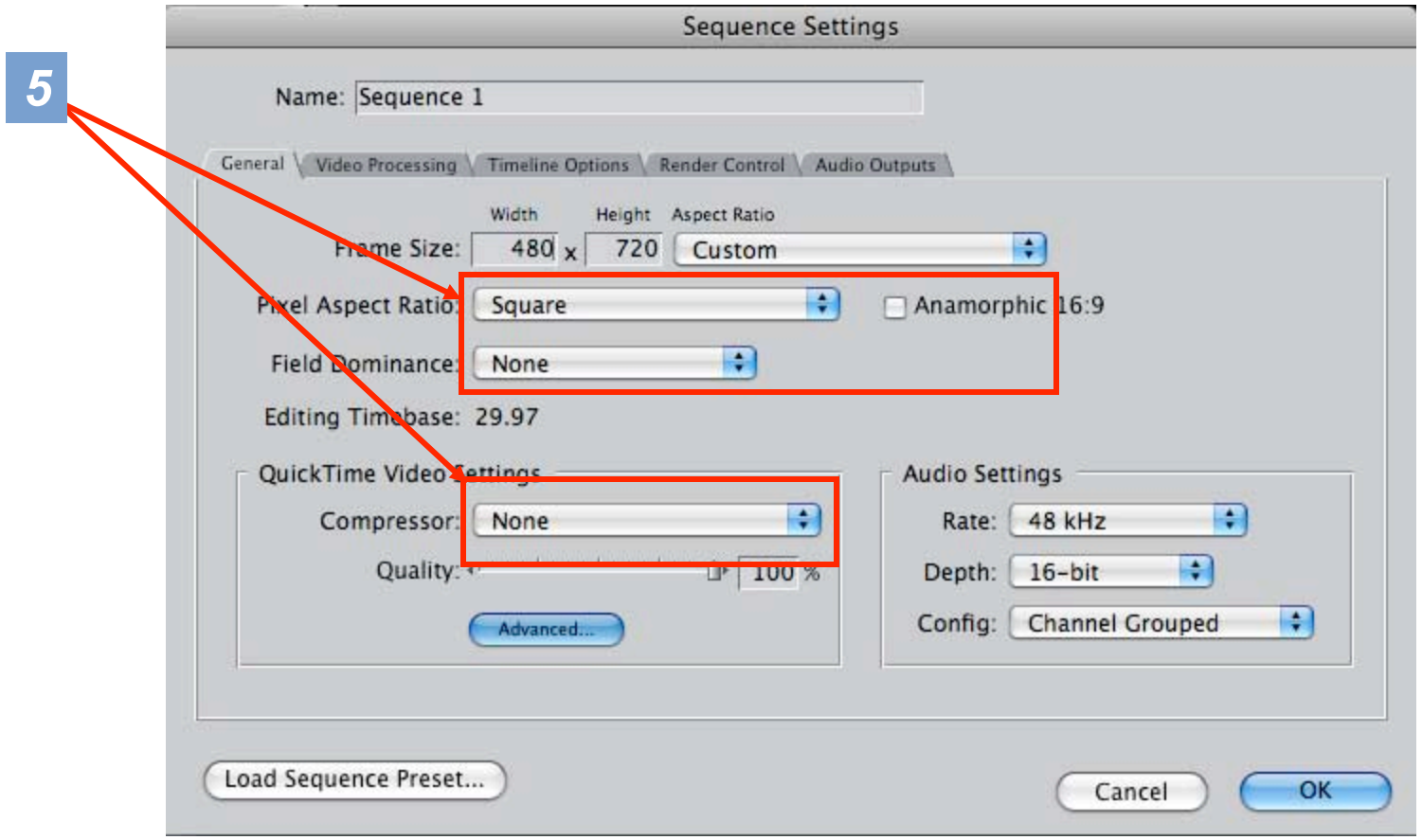
Custom Framing Vertical Video: in FCP

- ***Step 4*: Enter the desired frame size using > "Custom".**



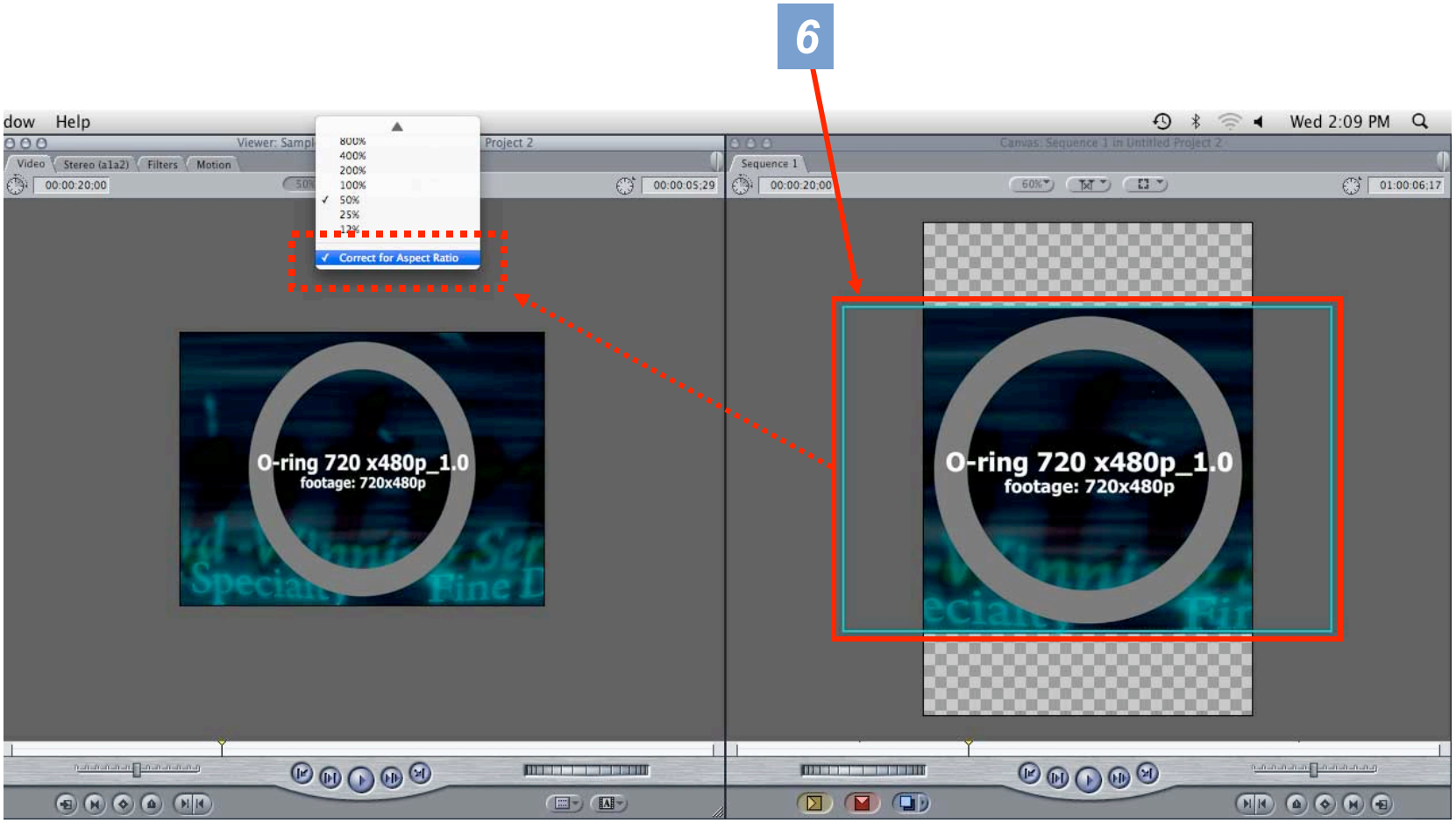
Custom Framing Vertical Video: in FCP

- ***Step 5*: Match these settings using the drop down menus.**



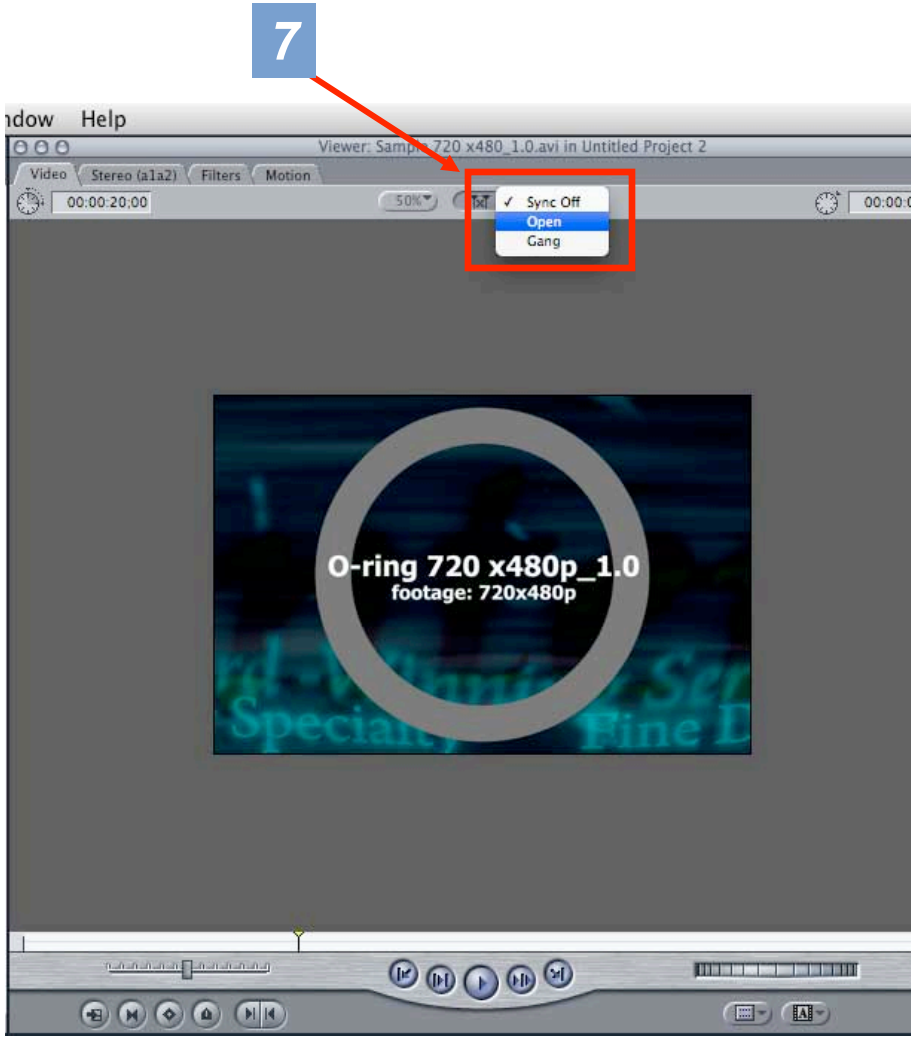
Custom Framing Vertical Video: in FCP

- ***Step 6*: Double-click** *the clip in the "Canvas" pane to load it into the "Viewer" pane, then uncheck "Correct for Aspect Ratio".*



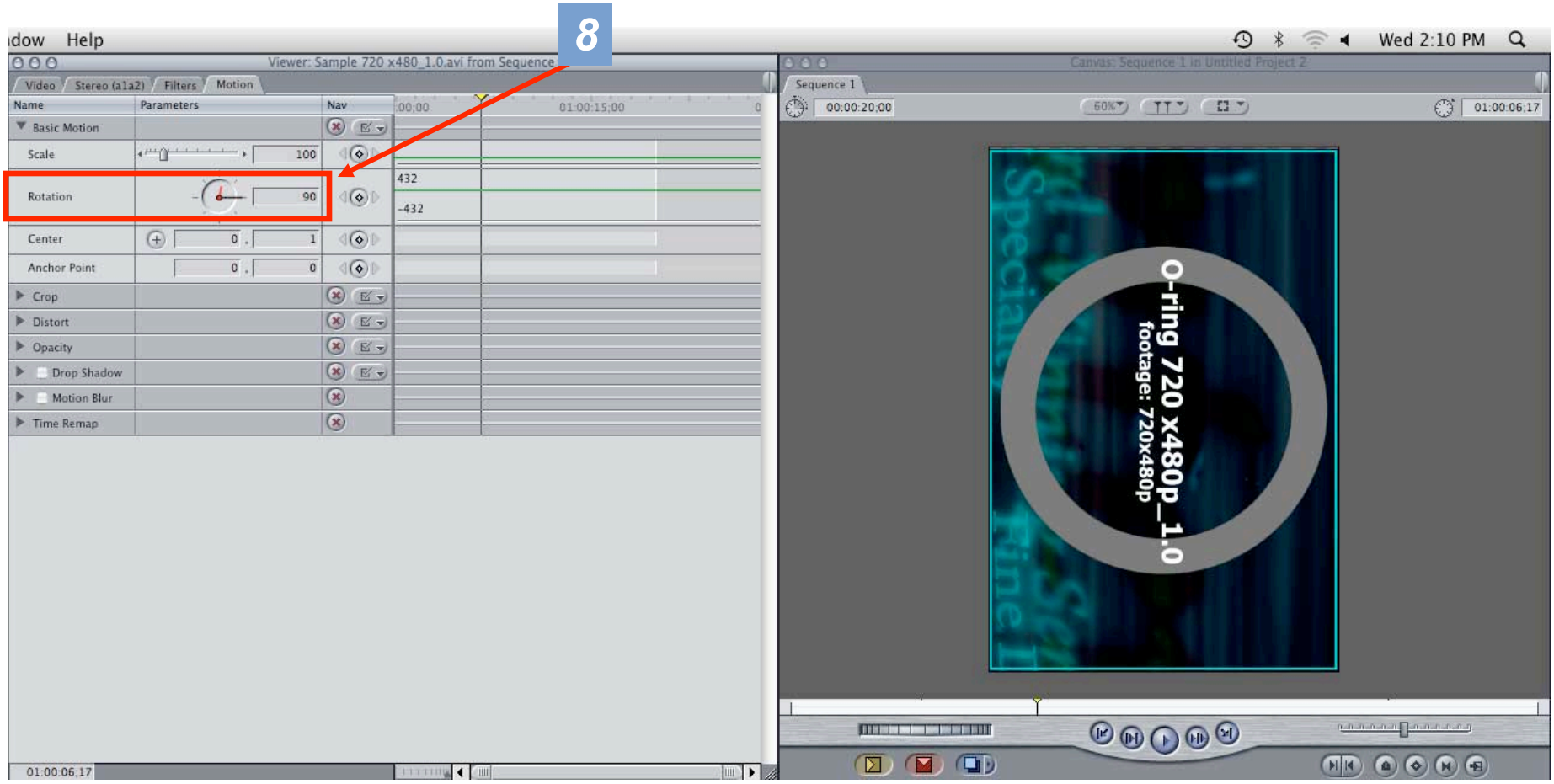
Custom Framing Vertical Video: in FCP

- ***Step 7*: Set** *"Synch" to "Open" to update the canvas.*



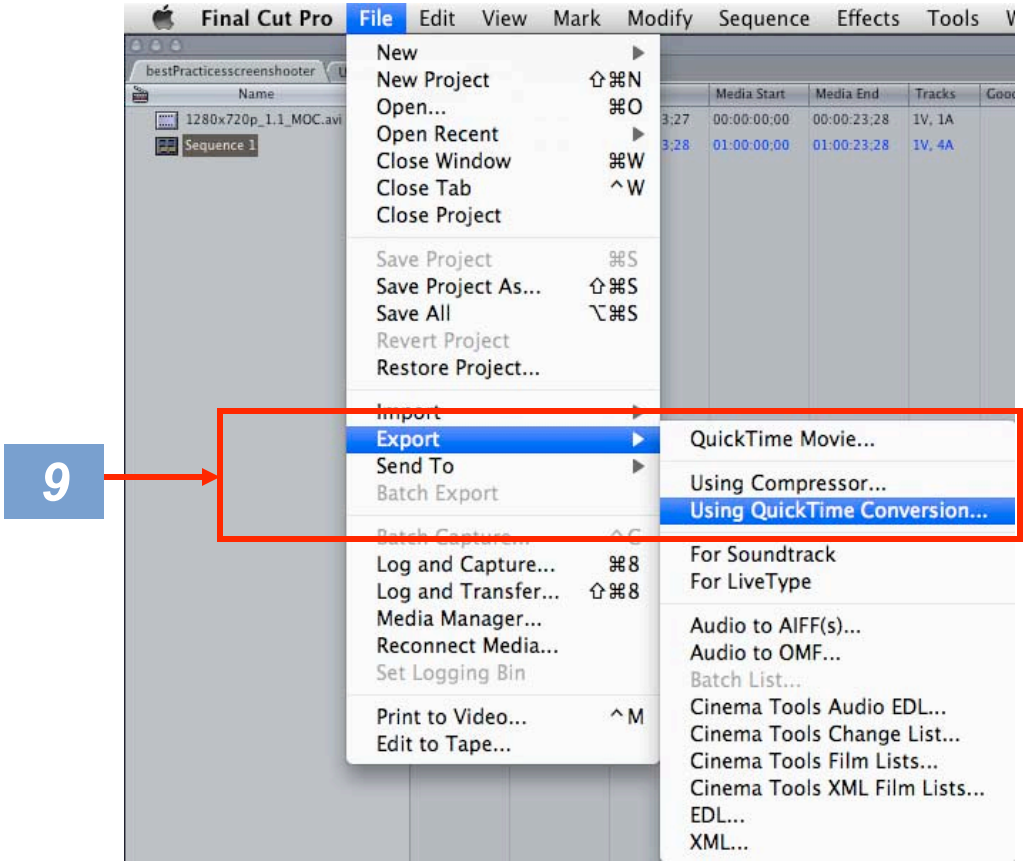
Custom Framing Vertical Video: in FCP

- ***Step 8*: In the Viewer's "Motion" pane, rotate the video 90 degrees.**



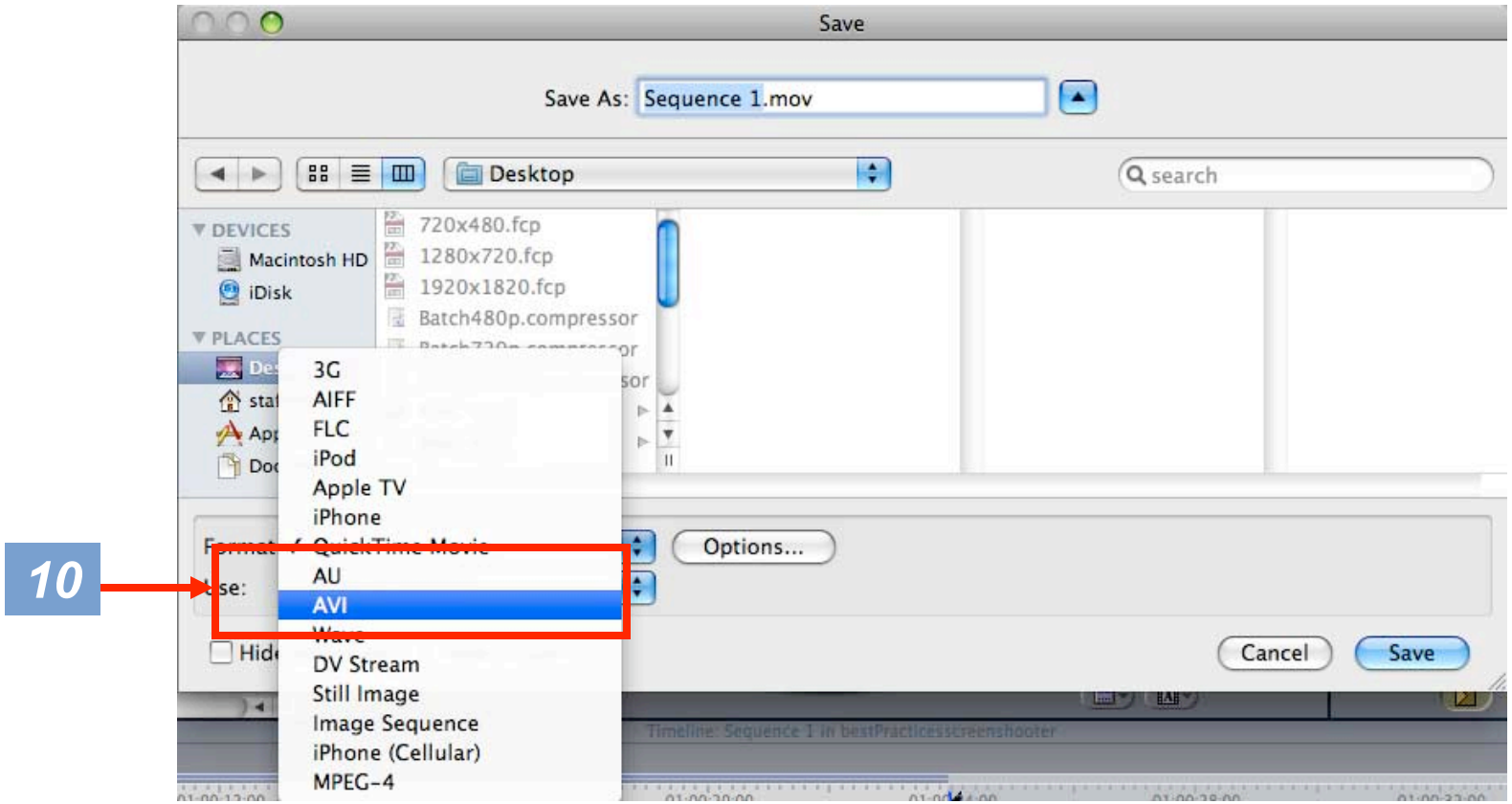
Custom Framing Vertical Video: in FCP

- ***Step 9*: File > Export > Using Quicktime Conversion…**



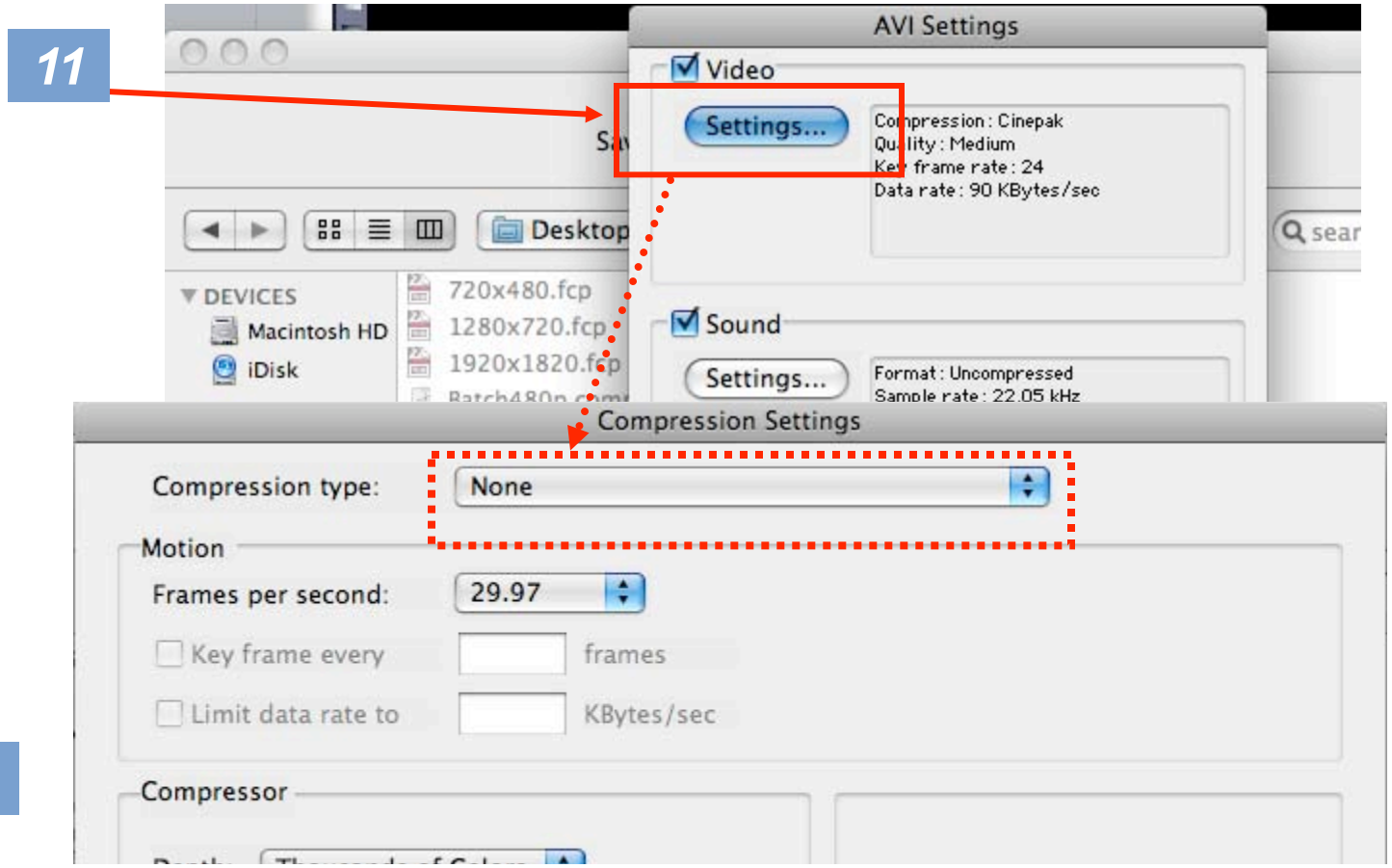
Custom Framing Vertical Video: in FCP

- ***Step 10*: Select AVI,** *then click "***Options***".*



Custom Framing Vertical Video: in FCP

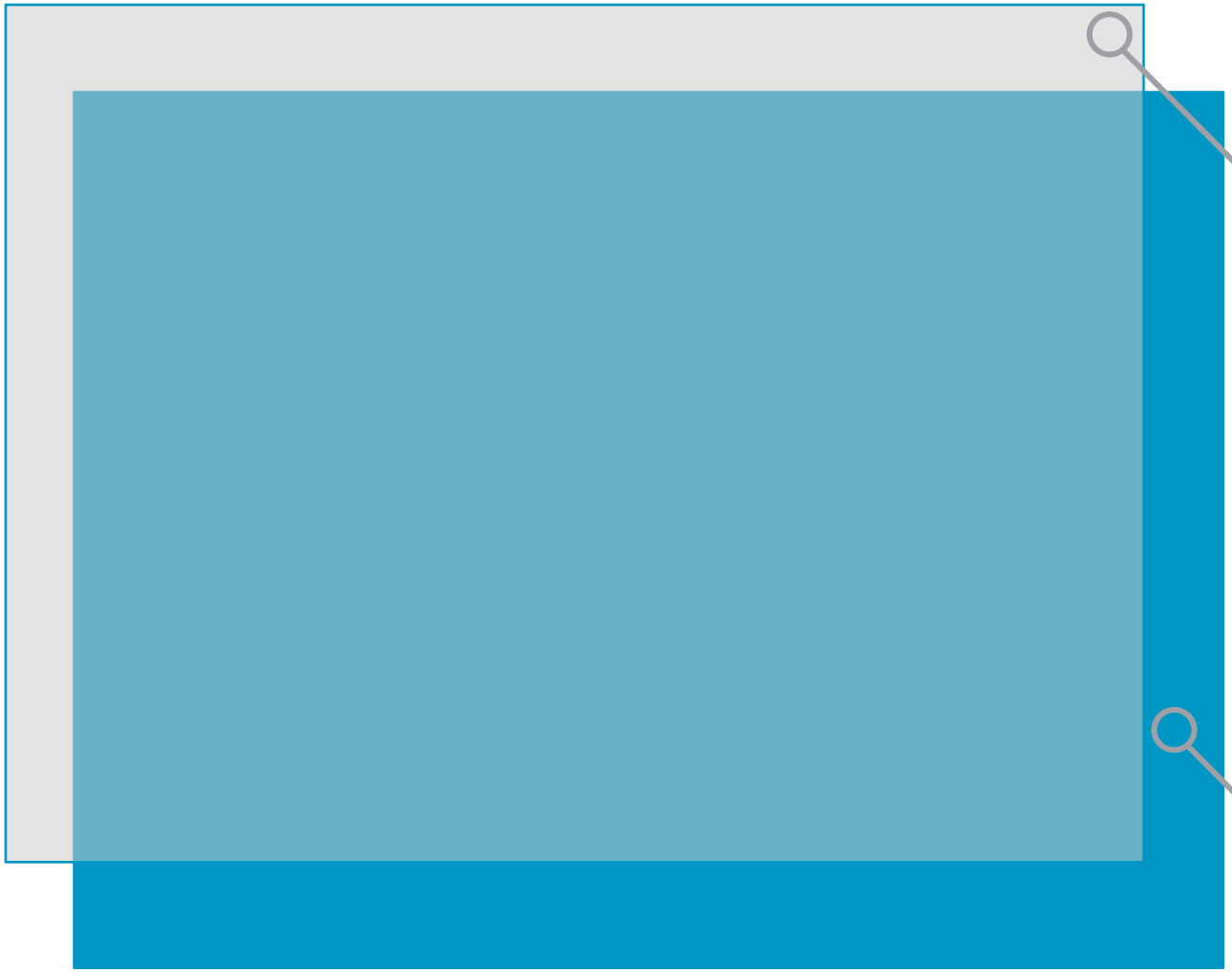
- ***Step 11:*** AVI Settings > Settings…: set compression to "**None**".



- ***Step 12:*** *Use the AVI video to generate an MPEG2-TS file.*

Applying Transparencies using API

Video and Graphics Planes



Transparent overlay example

Applying Object Transparencies Using API

```
mp.osdSetAlpha(window, 0, 0, window.screen.width, window.screen.height, 255);
    var el = document.getElementById("movie");
    var _el0 = document.getElementById("img1");
    var _el2 = document.getElementById("ani");
        if (el)

        mp.setOutputElement(window, el, 0);
          mp.setAlphaForElement(window, _el0, 160);
          mp.setAlphaForElement(window, _el2, 230);
```

OSD = On Screen Display

OSD = 0 -*just movie playing*

OSD = 255 -*web page elements*