CREATING SCORM-COMPLIANT ELEARNING COMPATIBLE WITH HTML, SUCCESSFACTORS, & MULTIPLE BROWSERS

Abstract

Many instructional designers are unfamiliar with the features and limitations of eLearning software and Learning Management Systems.

This document will explain the proper settings and best practices to use when creating and distributing eLearning that will work across various platforms and browsers.

Daniel Shannon
Instructional Designer
eLearning Developer
LMS Consultant
shannon5th@hotmail.com

INTRODUCTION: AUDIENCE AND LEARNING OBJECTIVES

Who is this instruction for? Instructional designers, familiarity with Captivate from intermediate to proficient who need to be aware of constraints imposed by…

- Citrix
- HTML5
- Learning Management Systems,
- SCORM, and/or
- various operating systems and browser versions used throughout Weill Cornell Medicine facilities.

Upon completion of this instruction, learners will be able to…

- optimize eLearning for use with Citrix system…
- create HTML5-compliant eLearning lessons in Captivate by recognizing, avoiding, and working around HTML5 constraints while creating Captivate lessons…
- use best practices while developing to optimize compatibility across operating systems and browser platforms…
- recognize and use optimal settings and naming conventions to ensure compatibility with SuccessFactors (SF)…
- understand SCORM (Shareable Content Object Reference Model) well enough to
  - use optimal settings to ensure SCORM-compatibility within SuccessFactors…
  - make informed decisions, diagnose and respond to potential issues going forward;

ELEARNING <> LMS PROCESS & PROBLEMS:

How do eLearning lessons work with the LMS to provide a learning experience & what obstructs the process?

How do eLearning lessons work with the LMS to provide a learning experience?

An eLearning event involves a series of interactions: file requests and data transfers.
- A learner clicks a button or enters information to create a message and waits for the server to respond…
- The message is transmitted through the browser to the Learning Management System…
- The LMS that the stores the eLearning image files, audio clips, videos, interactions, etc., receives the message and responds by sending all the images involved in the request through the browser and back to the learner…
- The lesson the learner is using acknowledges the reception of the files with a message to the LMS and processes the next interaction.
- **Depending on the settings of the source file in Captivate, interactions can trigger the transmission of dozens or hundreds of files constantly.**
- **Each transmission takes time: more time is needed for larger file sizes.**
- **At WCM, process is complicated further by number of different browsers, operating systems, and kinds of computers being used.**

Learner in eLearning lesson	Citrix Thin-Client System	Browser (IE / Safari / Chrome / Firefox)	Operating System (Windows / Mac)	SCORM-compliance	SuccessFactors LMS on server (stores folders, files)

Learners open lesson…

Server begins sending all files that were zipped up in folder…

While server is still sending, learners interact w/ files (clicking images, listening to audio, playing video, etc.) …

Server records every interaction …

Learners advance to next slide…

Server responds…

Lesson acknowledges server recording interaction…

Server records interaction …

Learners answer quiz questions…

Server records interaction …

Learners fast forward through video…

Video resets for each learner…

Potential Problems:

1. Timing: Each step in the process takes time. What increases the time required?
   a. File sizes (Captivate objects)
   b. More requests for files / interactions = more time spent sending information back and forth.
   c. All servers, not matter how large or fast, can be overwhelmed by excessive demands placed on them.
   d. **Excessive demands cause the learning experience to lag for the user while the files and requests buffer, sometimes to the point where the system freezes up.**
   e. Thin-client systems, such as Citrix, are especially limited when it comes to rich media files (see below).
2. Kinds of files:
   a. To work across all (or at least most) browsers, it is highly recommended that all files are HTML5-compatible. Essentially, this means no Flash animations or videos and no swfs (small web files).
   b. Video files: In addition to video file size, the kind of video used may be critical.
3. MyApps (Citrix):
   a. Although you may not be using a thin-client computer station, using MyApps through the Citrix system will only be using the server's memory, not your computer's resources.
   b. The limited bandwidth presents a challenge for video-heavy lessons. See footnote:1

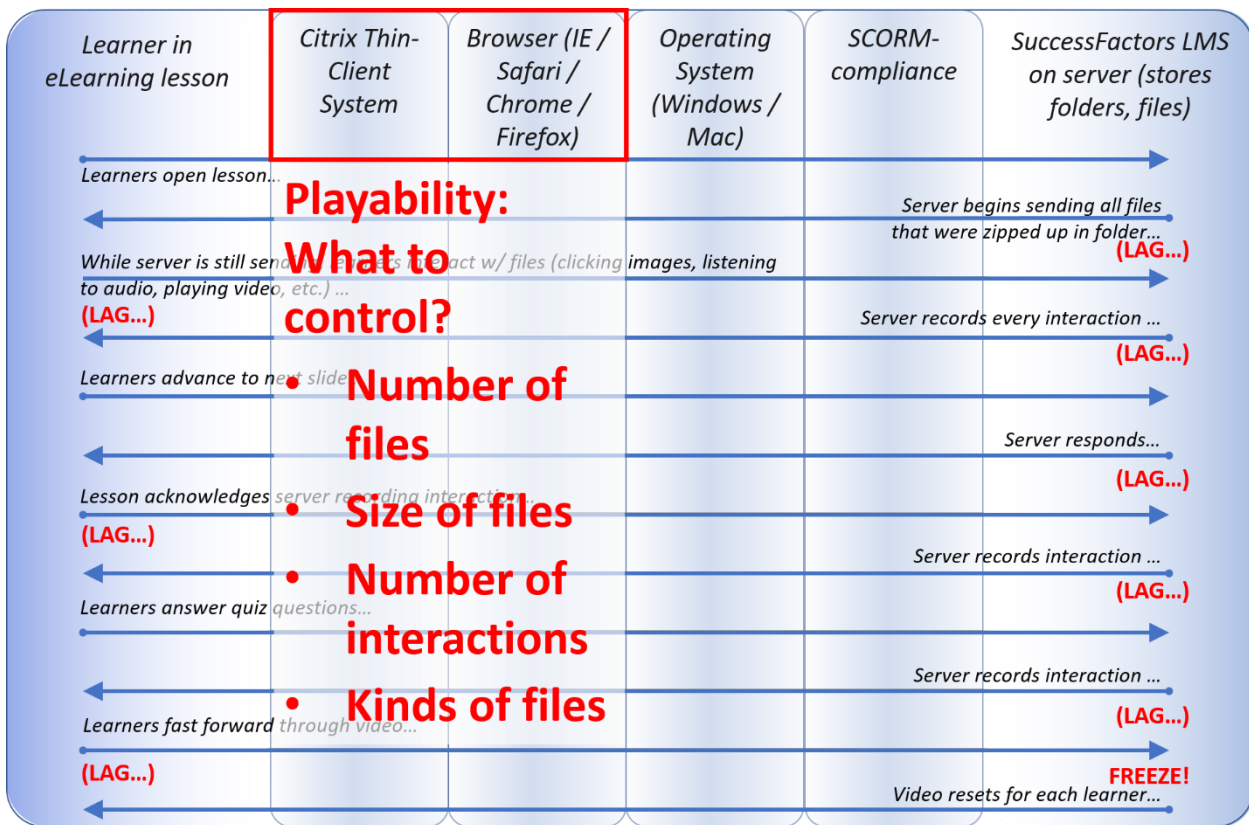[1] **"The Disadvantages [of thin-client systems]:**

"Client" Organizations are Subject to Limitations – Inasmuch as the slim units do most of their processing at the server, there will be setups where rich media access will be disabled. Some of these concerns are the result of poor performance when simultaneous access to multimedia in this PC is taking place. Heavy and resource-hungry applications like Flash animations and video streaming can slow the performance of both the server and the "client". In corporate organizations where video conferencing and webinars are often carried out, presentation of materials and web-cam/video communications can be adversely affected."

    c. All the video, audio, images, interactions, etc., contained in your eLearning lesson is constantly being swapped back and forth without the help of your individual computer's resources. Therefore, thin-client systems have a harder time dealing with videos than desktops and laptops.

4. <u>Working across various browsers & operating systems</u>
    a. Minimizing file sizes and interactions for use in Citrix should help eLearning go smoothly across various browsers and operating systems.
    b. HTML5-compatibility should help ensure that your eLearning can be used in maximum number of browsers and systems.
    c. MP4 video formats are about as universal as any video format: finding the best MP4 format/encoding may take some experimentation.

5. <u>Communicating with LMS via SCORM</u>
    a. The zipped package published by Adobe Captivate must be SCORM-compliant, according to specific SuccessFactors requirements.
    b. To achieve SCORM-compliance, the eLearning must be able to send a signal to the LMS that it has been completed, either through
        i. a quiz that has been passed,
        ii. an attestation slide (basically a true/false quiz), and/or
        iii. the number of slides viewed (not recommended – more below).

6. <u>Using SuccessFactors naming conventions</u>
    a. Naming your work consistently in Captivate, while not critical, can help tremendously with using and finding your files when uploaded to SuccessFactors.
    b. SuccessFactors won't accept zipped folders with swfs.
    c. NOTE: The following cannot have spaces in their titles:
        i. zipped folder published by Captivate;
        ii. Course and SCO identifiers in the Manifest configuration screen cannot have spaces either (more later).

CITRIX / BROWSER ISSUES: How can we create the most "playable" eLearning in Captivate to accommodate unavoidable limitations of servers / bandwidth?

Overview of Captivate eLearning Development Process

If you already know how to develop interactive eLearning in Adobe Captivate, then ensuring your lessons are SCORM-compliant, html-compatible, and usable in SAP SuccessFactors should not present too much of a challenge.

---

http://www.brighthub.com/environment/green-computing/articles/66417.aspx

Learner in eLearning lesson	Citrix Thin-Client System	Browser (IE / Safari / Chrome / Firefox)	Operating System (Windows / Mac)	SCORM-compliance	SuccessFactors LMS on server (stores folders, files)

Learners open lesson…

Playability: What to control?

Server begins sending all files that were zipped up in folder…
(LAG…)

While server is still sending, learners interact w/ files (clicking images, listening to audio, playing video, etc.) …
(LAG…)

Server records every interaction …
(LAG…)

Learners advance to next slide…

- **Number of files**

Server responds…
(LAG…)

Lesson acknowledges server recording interaction
(LAG…)

- **Size of files**

Server records interaction …
(LAG…)

- **Number of interactions**

Learners answer quiz questions…

Server records interaction …
(LAG…)

Learners fast forward through video…

- **Kinds of files**

(LAG…)

(LAG…)

FREEZE!
Video resets for each learner…

Lagging data transmission, buffering, and system freezes are caused when the servers can't accommodate the requests sent back and forth from the user to the LMS.

How can we design the eLearning to reduce the time it takes to process requests and ensure the eLearning experience proceeds without lags or breakdowns?

We can do several things:
- minimize the number and sizes of files being requested;
- minimize the number of interactions available to the user (without compromising instructional impact);
- **revise settings to minimize the number of interactions between the eLearning lesson and the server (this will probably have the biggest effect).**

What is Adobe Captivate, fundamentally?

Adobe Captivate is an eLearning development tool that can present images, text, audio, video, and specific kinds of learning interactions in various arrays and sequences programmed by instructors.

Learners can be offered choices in how to navigate a learning experience. Effective learning experiences can (and should) be interactive: users can and should be required to respond to events presented by eLearning developer.

Captivate Objects

Essentially, Adobe Captivate lessons are composed of basic objects such as

Captivate objects (files):
Building blocks of eLearning

Images   Audio   Text
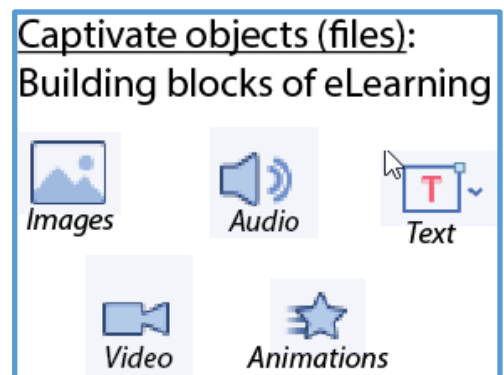
Video   Animations

- image files,
- audio clips,
- text boxes,
- video clips, and
- animations.

Videos clips are the largest kind of object and present the greatest load on the server / learning management system.

Using the lowest acceptable size and resolution for all objects is always good practice, but reducing the size and number of video clips is critical to ensuring your eLearning will play as intended.

- **Do not create videos of screen events in other screen capture programs and import these into Captivate slides as videos.**
    - Instead, use Captivate's screen capture capabilities.
    - If there are problems, you can
        - simplify the screen capture process within Captivate (the more complicated automatic screen captures often don't work perfectly) and/or
        - capture static screenshots with another program like SnagIt or the Snipping Tool and import these separately.
        - You can manually add click boxes, highlight boxes, mouse movement images, etc. and arrange them on the timeline to create the desired effect.
        - Screen captures done this way are more manual than automatic, but you will have better control of the interactions in your presentation.

Captivate objects (files): Building blocks of eLearning

Images   Audio   Text

Video   Animations
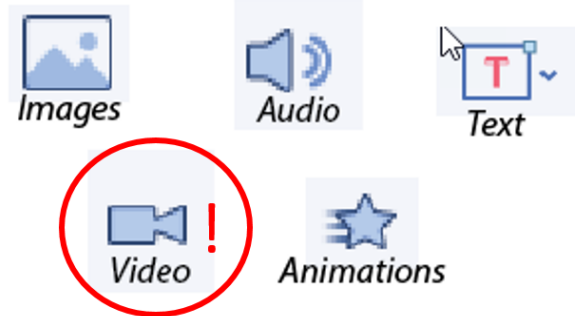
Minimize size:
- Duration
- Resolution (anything over 1000 pixels on a side is probably overkill).
- Frame rate: 30 fps is standard. Lower is better.
- Bit rate: 1.5 is probably the maximum to use.

- With practice, you may find yourself working more efficiently in a more manual mode than attempting to use the automatic screen capture features and revising your work afterward.
- **Example:**
  - Videos showing software simulations imported into Captivate are three to ten times larger than software simulations created within Captivate, creating bottlenecks in transmission at the Citrix level.
  - Additionally, the kinds of video files created may not be compatible across all browsers and operating systems.
- **UPSHOT:**
  - Use MP4 videos only.
  - Minimize number and size of videos where possible.

1: You can adjust settings to meet your needs. I've found that automatically adding text captions is more trouble than it's worth. Adding Click Boxes and Highlight Boxes can be helpful, but they usually have to be adjusted manually to fit.

- Be aware that the kind of MP4 video you're using can affect playability. Some experimentation may be required to determine the most consistently playable video format across browsers and operating systems.
- You may have to re-encode MP4 video clips to reduce
  - Resolution: Pixel heights and widths of over 1000 may be higher quality than is necessary.
  - Frame rate: 30 frames per second (fps) is standard, but slower frame rates will probably work as well in terms of instructional impact.
  - Bit rate: 1.5 MB/second is probably the upper limit. Lower is better.

Audio clips are the second largest kind of clip.

The Captivate source file can store all the imported or recorded audio files at high quality. The developer can publish the audio at the lowest acceptable quality to reduce file size.

So, go ahead and record the audio with the highest quality setting: you can always reduce it as needed.

Images: Best practices are to re-use, resize, and optimize image size wherever possible
- Each separate image imported onto your screen is a separate object to be processed.
- If you copy and paste a corporate logo onto each page, it will be treated as a separate object.
- Instead,
  - Capture images at the lowest acceptable resolution.
  - Store each image in the Captivate library and, if you need to re-use it, pull it from there.
- That way, instead of 30 logo images (one for each slide), the LMS will treat it as one.
- Similarly, use Master Slides for backgrounds wherever possible.
- Build directly in Captivate using Master Slides rather than importing from PowerPoint.
  - When you import from PowerPoint, each background image is a separate object.
  - When you use Master Slides, the background images are reused.

Sequencing

The order in which these objects appear to the user can be controlled completely by the developer. The developer can also allow the user to control the order in which the objects appear.

Because of the way the entire eLearning experience

must be transmitted to the learner, **it may be best to limit the size of the initial objects at the <u>start</u> of the lesson**. Some developers will use a few introductory slides, light in content, before sending videos. This gives the user / client computer time to receive all the files needed for the learning experience. They can work their way through the first slides while the video is transmitted.

Sometimes you'll find that your slides don't pause when you have set them up to. **To optimize playability, it's a good idea to leave a short space of time before and after each audio clip: a few tenths of a second is sufficient and learners will be unlikely to notice.**

Interactions

The developer can also control how the user will interact with the lesson and the server.

The number of interactions a developer deliberately programs or allows shouldn't present too much of a problem (unless the user clicks excessively).

However, Captivate is set up, by default, to interact with a Learning Management System after every single slide. Each click, each text entry, each quiz answer, etc., is transmitted to the LMS and then Captivate waits until the LMS has acknowledged the interaction before processing the next interaction (see above).
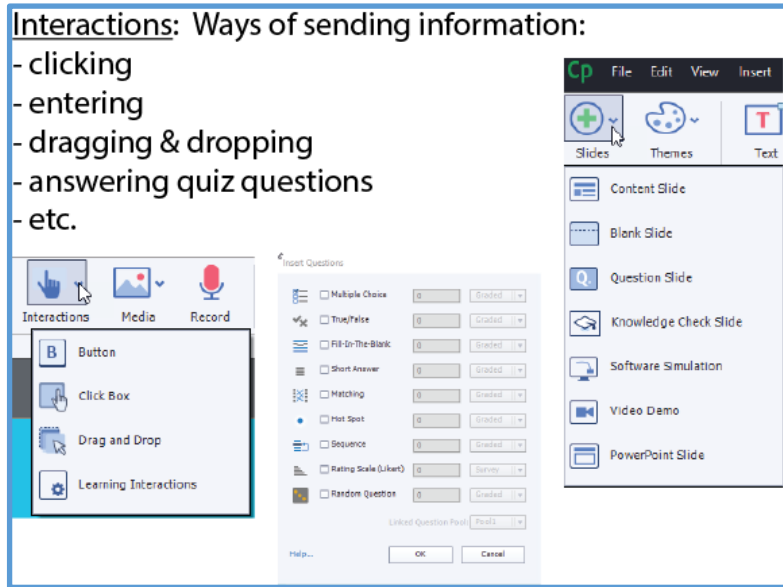
These default settings can…

- enable users to leave the program and the LMS will remember exactly where they were, among other advantages, **but**
- sending this much information back and forth every slide can lead to a frozen server, if more than a few people are taking quizzes or interacting with slides at the same time.

Reducing the number of default interactions is probably the best way to minimize the load on the server and help ensure efficient playability.

However, the benefit must be considered against the convenience of enabling users to return to the same slide they were working on when they left the lesson. Ways to diminish the inconvenience for users include:

- Using navigation that enables users to find the place they were working on when they left;
- Creating shorter eLearning experiences so that more learners can finish them in one sitting.
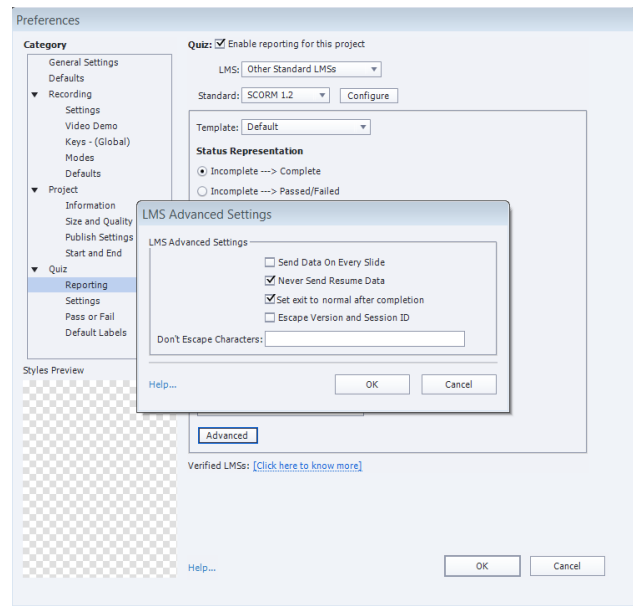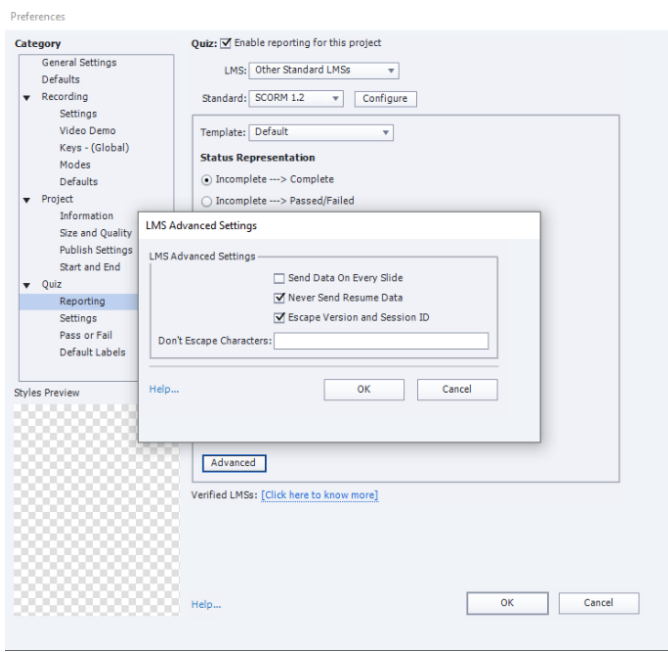
Another excellent way to reduce the number of unwanted interactions is by limiting or eliminating the playback options available to the user.
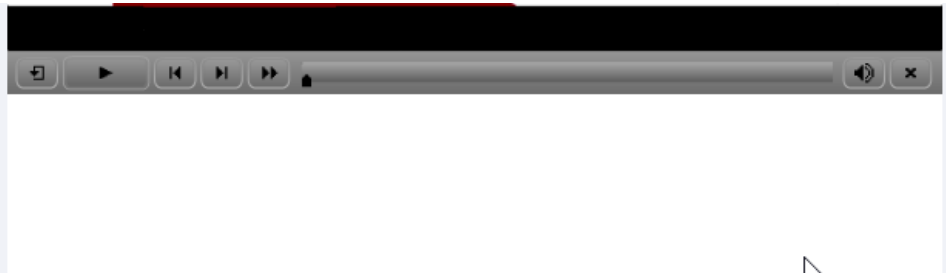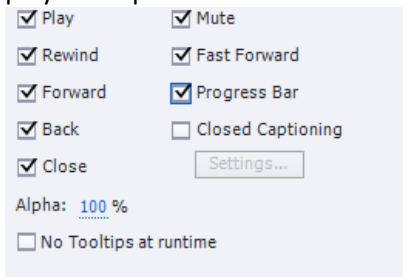
By default the playbar settings include:
- Play
- Rewind to beginning
- Forward (one slide)
- Back (one slide)
- Close (Exit the lesson)
- Mute
- Fast Forward (2x or 4x)
- Progress Bar (where you can skip forward to any point in the lesson)

The problem is that when a user clicks Fast Forward or uses the Progress Bar to jump around in the lesson, the server will probably not be able to keep pace. The audio, especially, fails to keep up and refreshing the entire lesson is sometimes necessary.

eLearning developers can control the playability of their lessons by creating their own buttons and eliminating the playback bar. If the developer delays the appearance or functionality of the **Next** button, the program will have more time to receive the files and process the preceding interactions.

At the very least, developers can de-select the **Fast Forward** button and the **Progress Bar** without negative instructional impact. Learners can still control their progress through the course, but they won't be able to make the big jumps that overload the server.

Check for lean design, development
- Sizes of files
  - Images
    - Reused where possible?
  - Video
    - Optimal kind, settings?
    - If video is used, is it sequenced later in the presentation to prevent buffering issues?
  - Audio:
    - Recorded at high quality, published at reduced quality?
- Interaction settings
  - Do we need to assess the number of slides the user has viewed as a completion criterion?
  - Are all the settings checked to minimize slide by slide interactions?
  - Check the playback bar for unnecessary and possibly disruptive buttons.

How can we make sure our eLearning is HTML5-compatible and that it works across as many browsers as possible?

1. <u>What is HTML5 and why do we have to use it instead of Flash or swfs when publishing?</u>
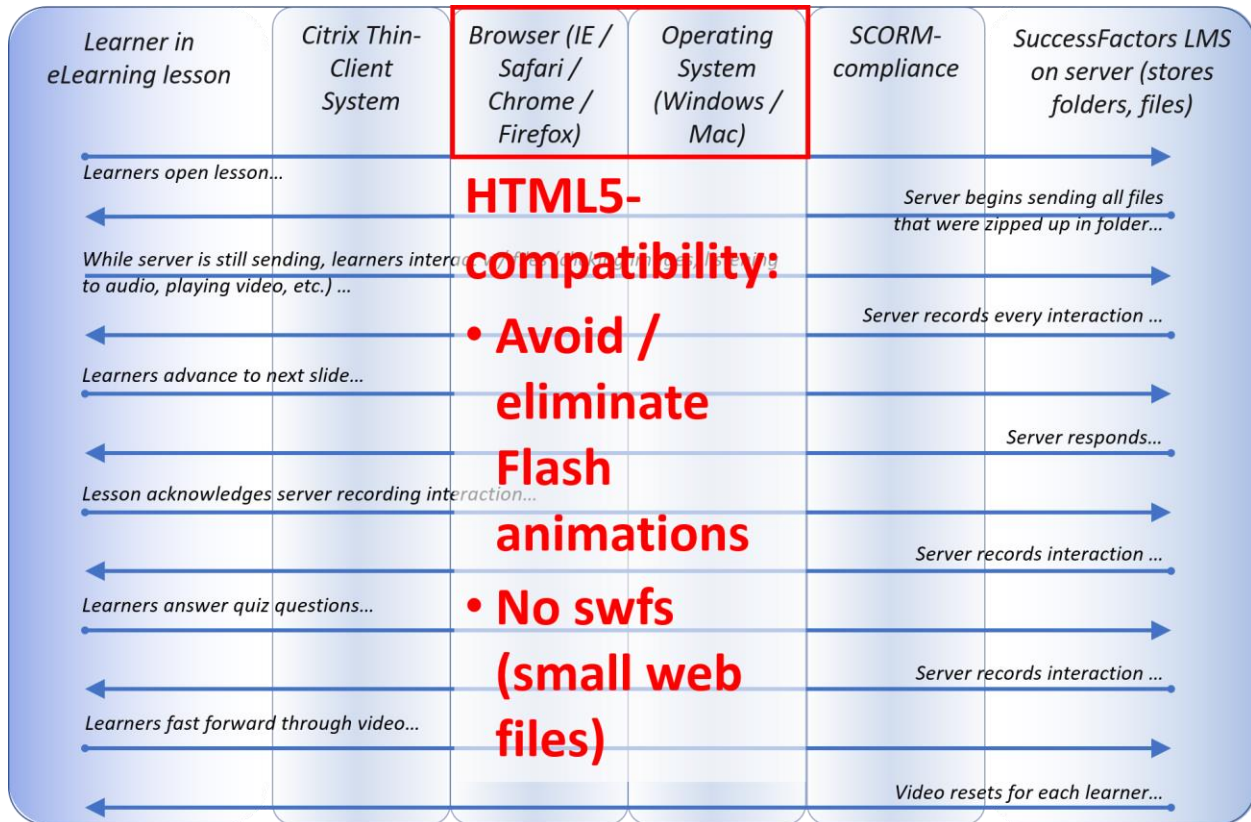
Captivate *can* publish your eLearning as either
- A zipped html5 folder,
- A single Adobe Flash file, or
- Both.

Single Adobe Flash files are great: they're small, versatile, and easy to deal with. Also, you can create nice special effects.

Unfortunately, due to security issues, Flash-based files and animations are not automatically supported by many browsers or by SuccessFactors. Although you may be able to get away with using Flash animations in your lessons in some browsers if you adjust the settings of individual computers, it will not work automatically in most. Moreover, SuccessFactors checks each zipped folder for swf files and will not import it if it finds any.

Instead of the single file created when you publish your source file as a swf, Captivate HTML5 output is a zipped folder, a collection of all the images, text, videos, and other files that comprise your lesson. Included in this folder are instructions for the LMS and the learner.

> **NOTE:** It may be possible to hide swfs from SuccessFactors and get your folder accepted, but it's not a good idea.

HTML5 is designed to be compatible with all desktop and laptop devices, tablets, and smartphones. **Any interaction that <u>requires</u> a mouse, such as a rollover or mouseover effect, is not *supported* in HTML5, but <u>it may still work in the near future</u>.** Rollovers, for example, still work (as of July 2017).

When a file or animation / interaction is "not supported" it can mean one of several things:

- The animation works for now (rollover / mouseover effects, e.g.) but there is no guarantee it will work in the future.
- The Flash file or interaction within the file will not work, but won't affect playability in general ("glow" effects, text animations, slide transitions).
- The file or interaction will cause the project to not work properly (swf files / certain videos).
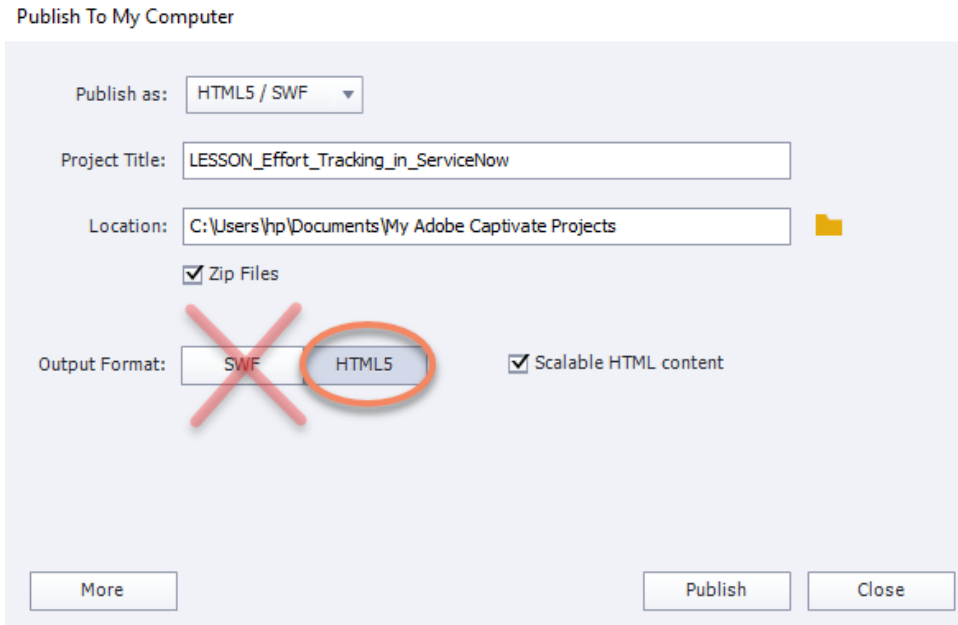
You can probably continue to use rollovers, but, for those who wish, there are HTML5-compatible work-arounds using Captivate's **Advanced Actions** with about the same functionality.
- Instead of moving the mouse over one image to see another, you can click one image to show another and at the same time hide the other images on the screen.
- You can also click one image to see a second image and click it again to clear it.

- Source files and examples for these two methods are included in the shared folder under **Captivate Information**.

Another best practice is to use only web-safe, HTML5-compatible fonts:

2. TESTING FOR HTML5 COMPATIBILITY

You can check for HTML5 compatibility with the **HTML Tracker** under the **Project** tab.

However, this tracker is not 100% reliable. It sometimes misses interactions that might cause problems and finds things that won't cause any problems.
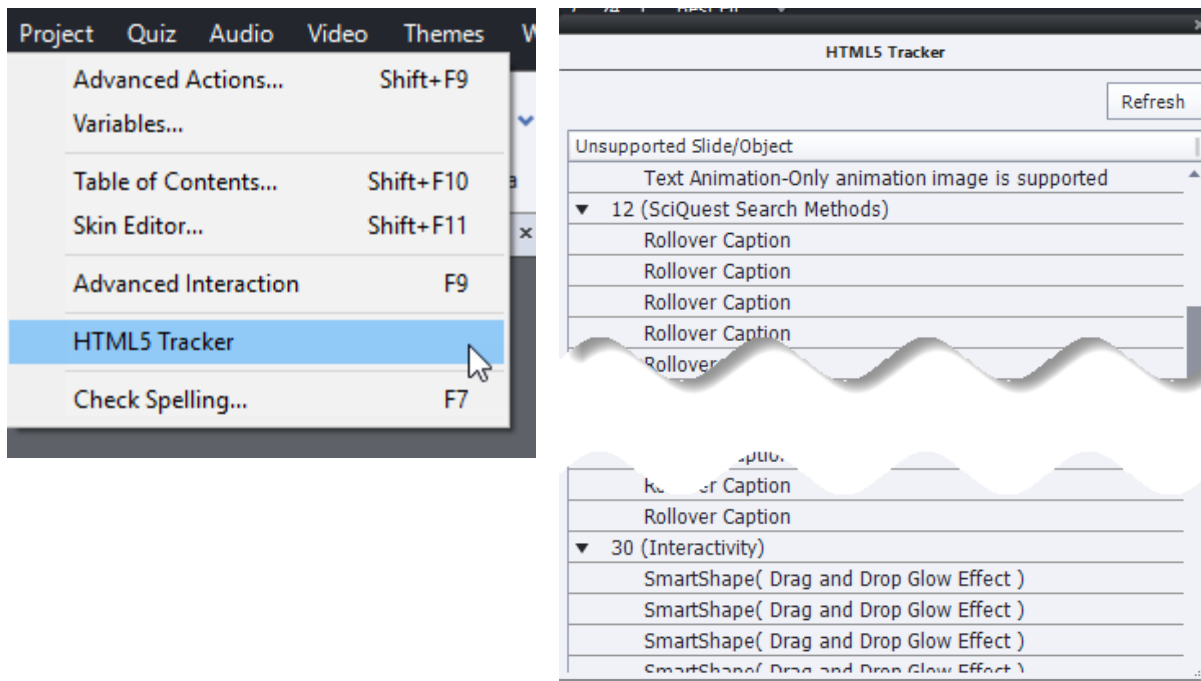
Objects in the project that are not supported in the HTML5 output get dynamically listed in the HTML5 Tracker. To view the tracker, click **Project > HTML5 Tracker**.

When you delete or modify such objects in the project, the tracker is updated.

The following objects and slides are not published in the HTML5 output:

- Text and SWF animations (only the first frame is visible).
- Mouse click animations (only one default click effect is supported).
- Question pools, Likert question slides, and random question slides.
- Slide transitions.
- Slide background if an SWF file is used.
- Audio attached to invisible objects.
- Borders.
- Reporting to an internal server.
- Video streaming using RTMP is not supported in HTML5 output.

Note: FLV files that are created using On2VP6 codec only are supported in the HTML5 output.

After using the HTML5 Tracker, you will still need to test your published output for any issues. There are several sites where multiple browsers are simulated.

SCORM ISSUES & WORKING WITH SuccessFactors

What is SCORM?
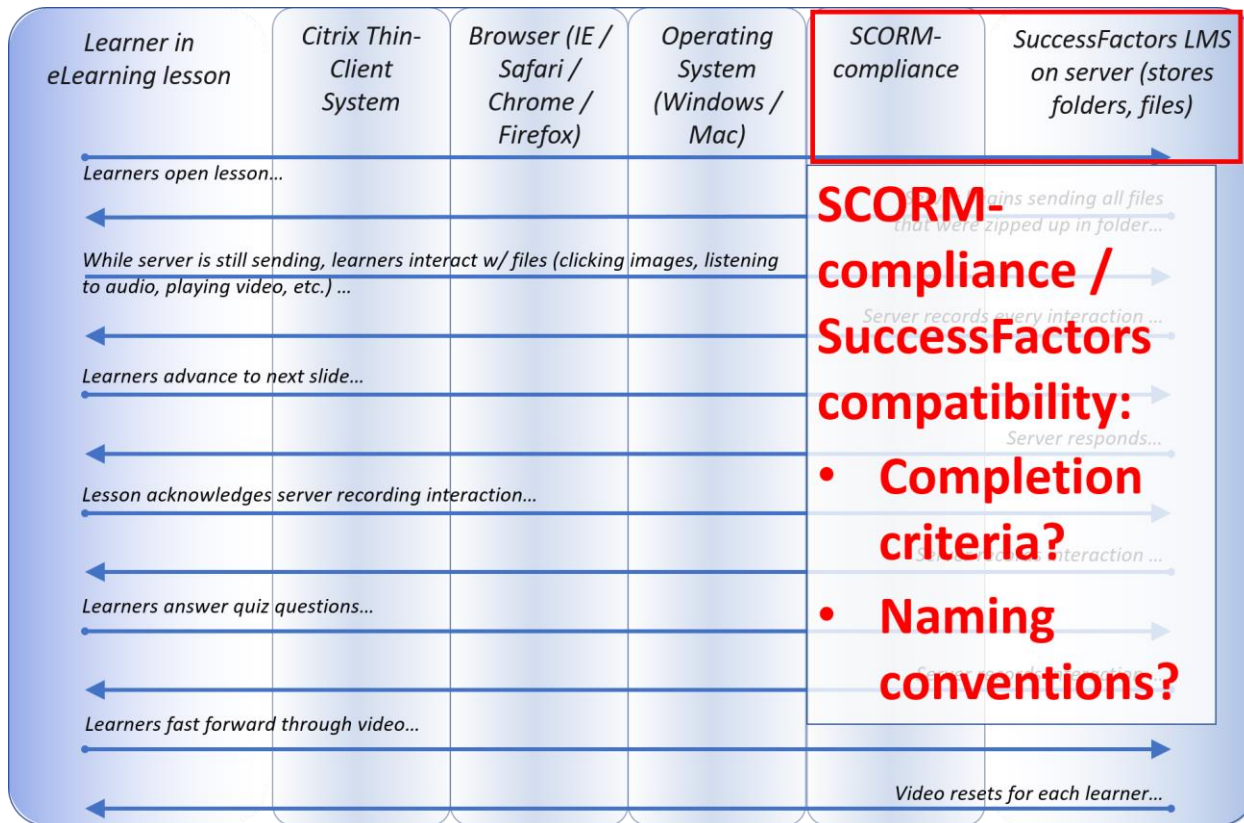
     SCORM is an acronym for **Shareable Content Object Reference Model**. There are different versions, but they all enable learning management systems to communicate with eLearning objects.

     SCORM enables you to create a Shareable Content Object (SCO, or just Content Object in SuccessFactors terms) within an eLearning development tool such as Adobe Captivate or Articulate Storyline.

     Basically, you can package your eLearning so that it cannot only be delivered to your learners as you intend, but also so that your learners can interact with it and send data back to you through a learning management system such as SuccessFactors.
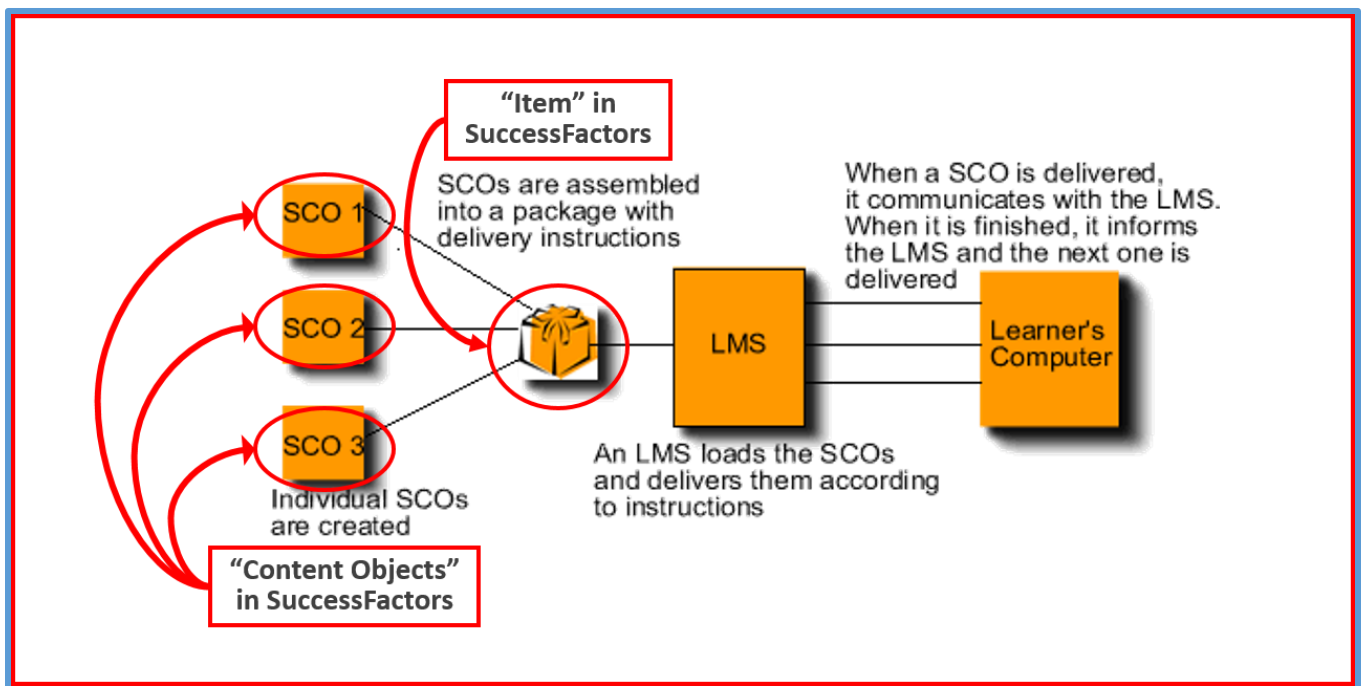
It's as if you not only create a package with specific delivery and opening instructions, but you also provide the recipient with the ability to send the package back to you with a specific message attached.

Each Learning Management System (or LMS) interprets the SCORM standards a little differently. Not all the settings we recommend for SuccessFactors are critical, but some are. This guide will explain what the various settings do, which are critical, which are advisable, and you can decide which you choose to use.

Making your lesson SCORM-compliant // SuccessFactors-friendly

In this section, you'll see how to use Captivate settings to ensure that your lesson is SCORM-compliant and works well within SuccessFactors.

To be uploaded to SuccessFactors, the zipped folder containing your lesson must be SCORM-compliant. That is, the zipped folder must contain information specifying the completion criteria for the course.

These settings can be defined after you've designed and developed your course.

At the same time, you can enter titles and identifiers for your lesson to enable it to be categorized and stored in SuccessFactors.

Determine your completion criteria for your lesson: To earn completion credit for the lesson, do you want your learners to …

- Just access the course?
- View all, or at least a specific percentage of, slides?
- Take a quiz but not necessarily pass it?
- Pass a quiz within a certain score range, within a certain number of attempts?
- View a specific number of slides and pass a quiz?

Let's assume we want users to pass a quiz in the Captivate lesson with a score of 75 or higher with just 3 attempts, but don't care if they view every slide. In fact, we've set up the navigation so that they have to spend a certain amount of time on every slide, so the issue can be resolved that way.

One more consideration: are there consistent naming conventions established in your organization that will allow your eLearning to be accessed within SuccessFactors? You will first add a Shareable Content Object to the SuccessFactors Learning Management System and then you will associate it with a course ("item" in SuccessFactors terms). It may be helpful to name the lesson in Captivate in such a way that it is easy to find and associate with an item/course later.

1) Click **Edit > Preferences**.	
2) Click **Quiz** to see your reporting options. 3) Click **Enable reporting for this project.**	
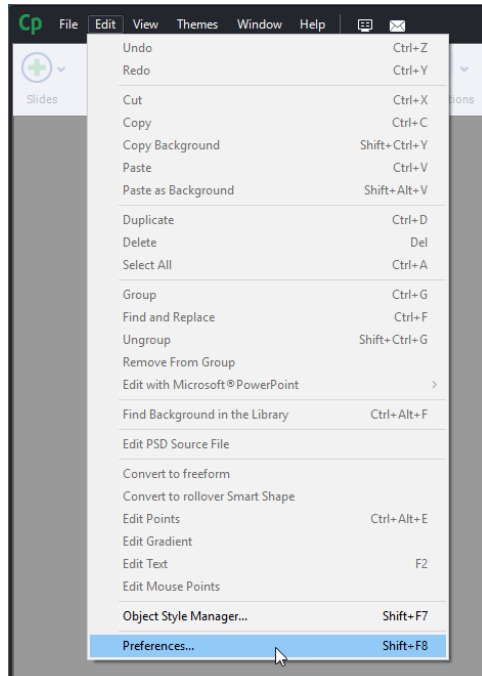
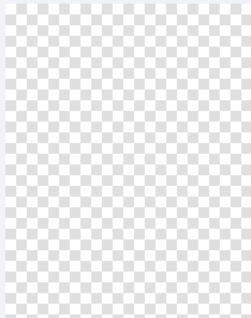|	
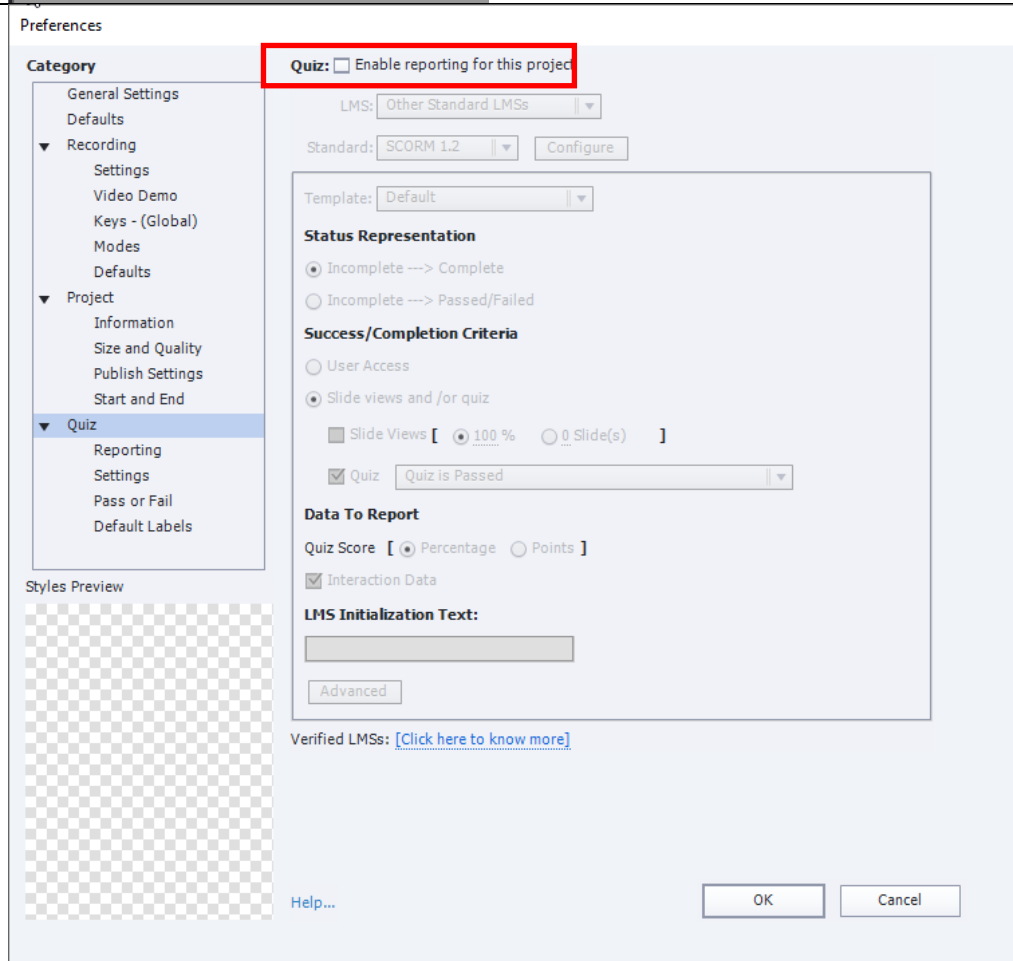
4) Note your options regarding SCORM, completion criteria, and other interaction settings.	**Preferences**
	
	Category
	General Settings
	Defaults
	▼ Recording
	  Settings
	  Video Demo
	  Keys - (Global)
	  Modes
	  Defaults
	▼ Project
	  Information
	  Size and Quality
	  Publish Settings
	  Start and End
	▼ Quiz
	  Reporting
	  Settings
	  Pass or Fail
	  Default Labels
	
	Styles Preview
	
	Quiz: ☑ Enable reporting for this project
	LMS: Other Standard LMSs ▼
	Standard: SCORM 1.2 ▼ Configure
	
	Template: Default ▼
	Status Representation
	⦿ Incomplete ---> Complete
	○ Incomplete ---> Passed/Failed
	Success/Completion Criteria
	○ User Access
	⦿ Slide views and /or quiz
	  ☐ Slide Views [⦿ 100 % ○ 0 Slide(s)]
	  ☑ Quiz Quiz is Passed ▼
	Data To Report
	Quiz Score [⦿ Percentage ○ Points]
	☑ Interaction Data
	LMS Initialization Text:
	
	Advanced
	
	Verified LMSs: [Click here to know more]
	
	Help... OK Cancel

5) Leave the screen on its default settings: **Other Standard LMSs and SCORM 1.2**. (Differences in SCORM versions are discussed below.)	

6) Click **Configure**.	

7) Note that the **Manifest** screen provides different ways to identify and describe the course.	Preferences

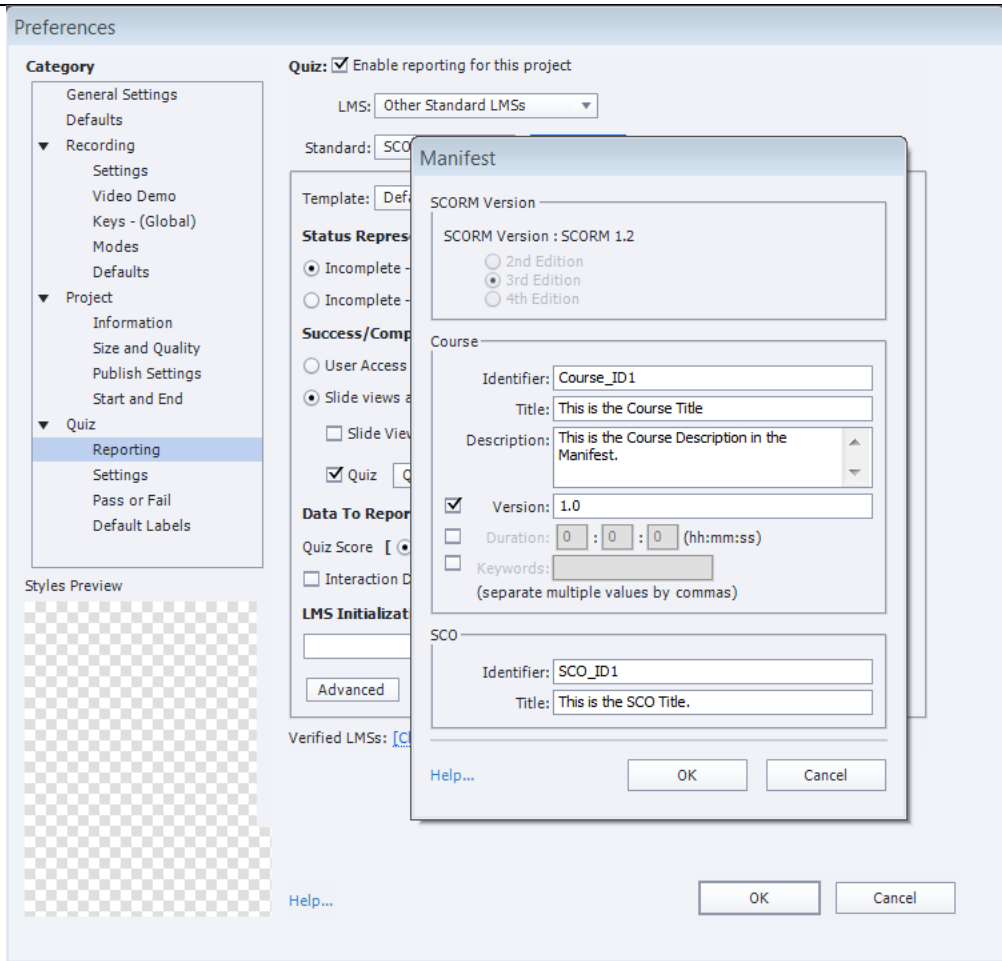|	
8) The **Identifiers** will provide SuccessFactors with a way to reference this SCO (Shareable Content Object[2]) in its system. The two 10-digit numbers will be combined and other numbers added to form a **unique identifier**. 9) **NOTE:** Leave these numbers alone: just make sure you don't add spaces or special characters to either **Identifier** field.	

---
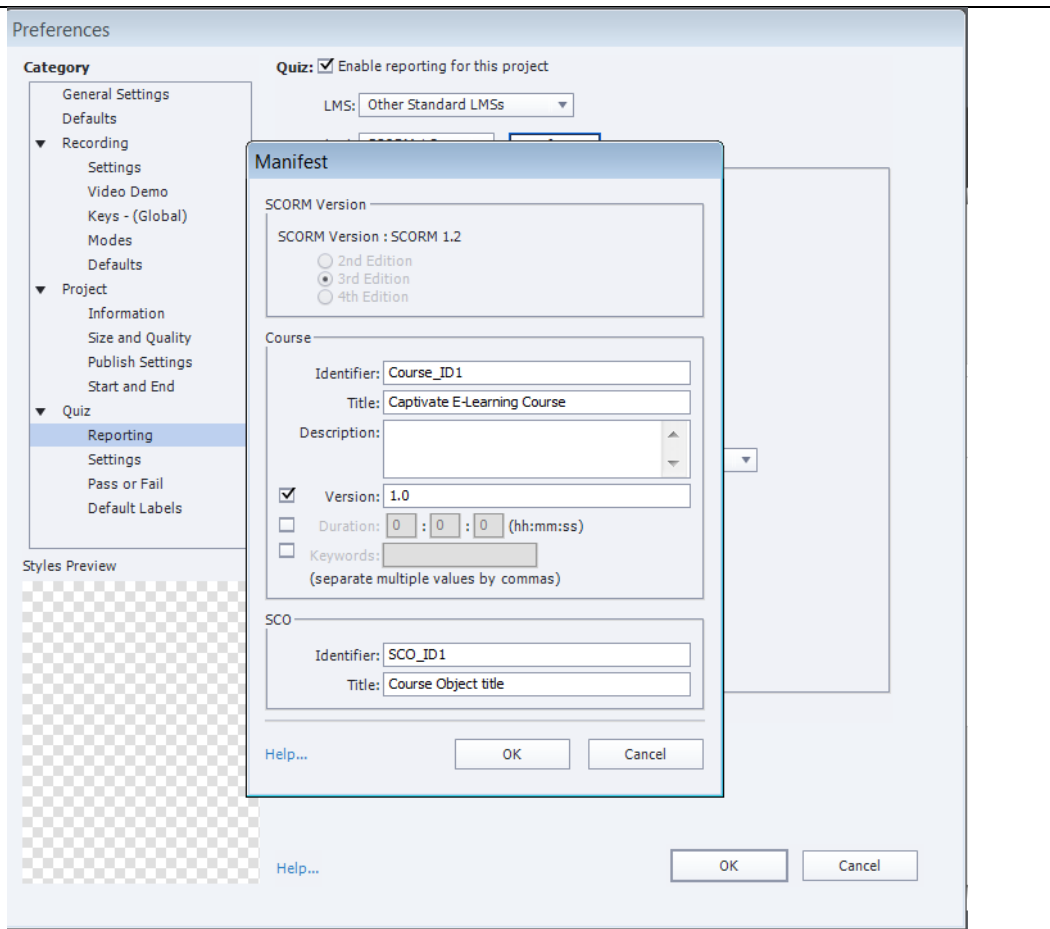
[2] Note: SCOs are called Content Objects in SuccessFactors.

10) The Course Title and the SCO Title are both important ways to identify your eLearning lesson now so that you can locate it later within SuccessFactors. 11) Each appears in SuccessFactors in different places.	
12) On the Item Details page, the **SCO Title** is used as the title of what SuccessFactors calls the **Content Object**. 13) The Course Description in the Manifest screen is used as the Course Description on the **Item Details** page and elsewhere throughout SuccessFactors.	
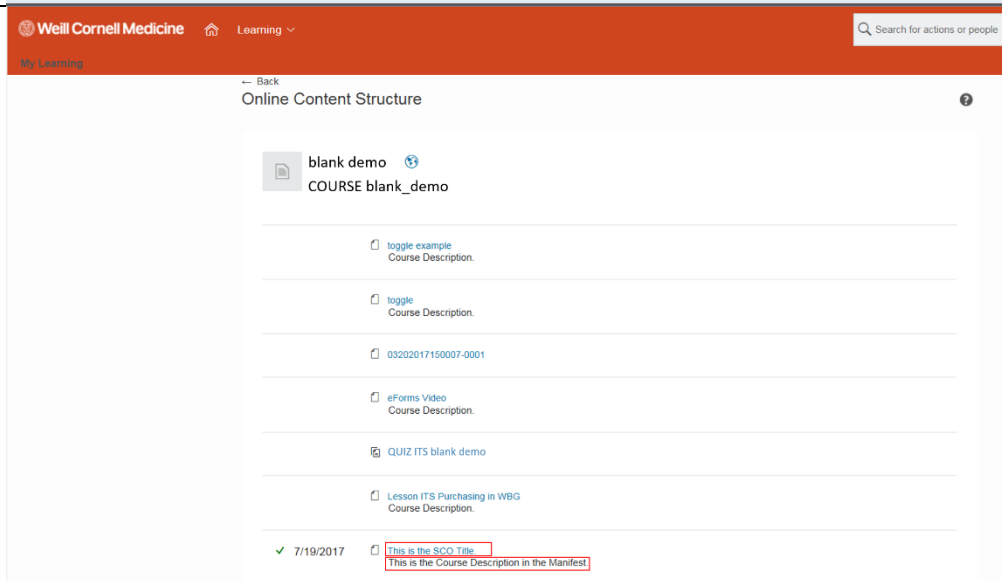
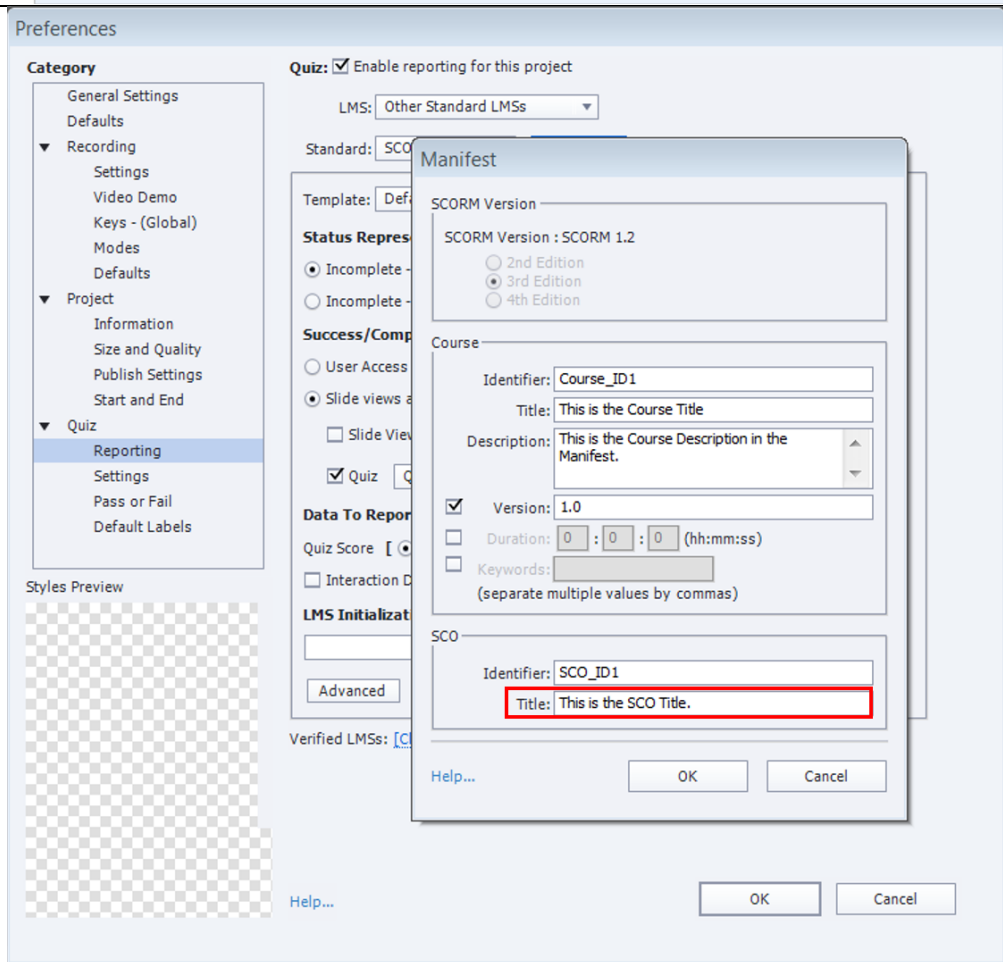|	
14) As a Learning Administrator, you'll also see the SCO Title when you list the Online Content for the course.	
15) Create an SCO title that relates to and can be associated with the course. 16) **NOTE:** The Course Title is used in some Learning Administration screens, but does not appear to the user often. 17) When you are ready, click OK.	
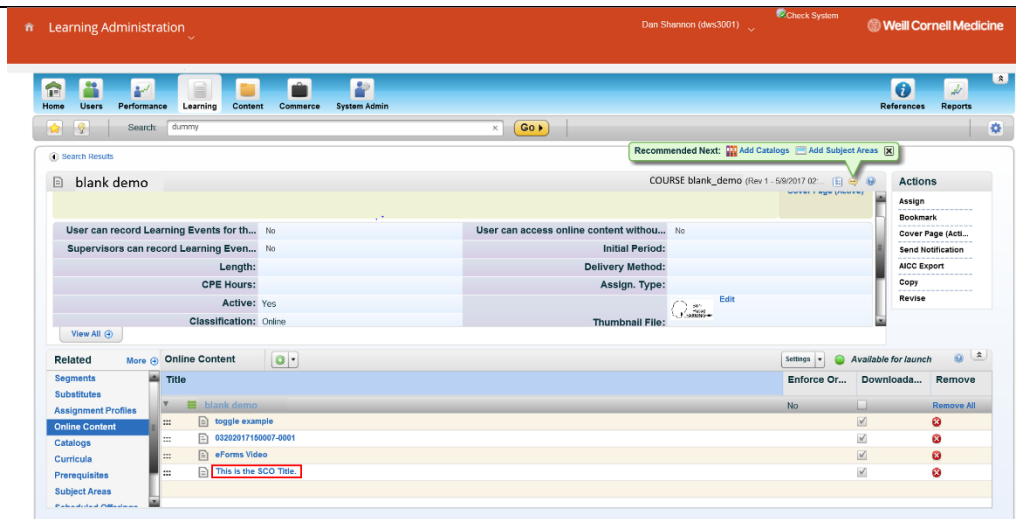
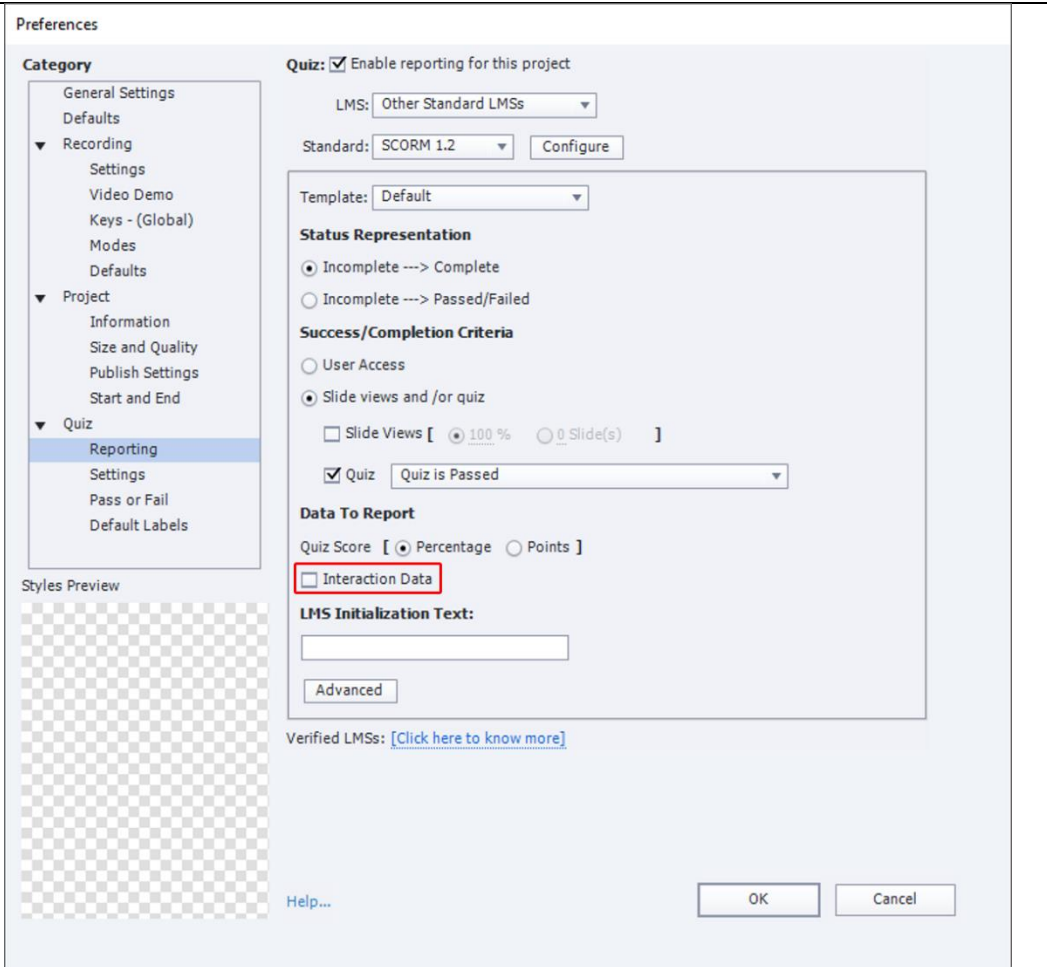|	
18) When users begin an eLearning, the "status" of the lesson for each user is incomplete until they fulfill the criteria established for either "Complete" or "Passed" or "Failed". 19) In SuccessFactors, "Complete" and "Passed" are handled the same way. 20) When a user is done with a course, the **Status Representation** is transmitted to SuccessFactors. 21)	 • When users begin an eLearning, the "status" of the lesson for each user is incomplete until they fulfill the criteria established for either "Complete" or "Passed" or "Failed". • In SuccessFactors, "Complete" and "Passed" are handled the same way. • When a user is done with a course, the **Status Representation** is transmitted to SuccessFactors.

22) You determine the Success / Completion Criteria: User passes / completes lesson when user - accesses lesson (**User Access**) - views specified number or percentage of slides (**Slide views**) (Significant load on server that must track every user's interaction with every slide). - attempts quiz, or passes quiz, or runs out of tries (**Quiz**) - **NOTE:** To be SCORM-compliant, the LMS needs some kind of completion criterion. That's why, if there is no quiz or criterion involving number of slides or user access, an Attestation slide consisting of a single question will provide the criterion needed.	Preferences Category General Settings Defaults ▼ Recording Settings Video Demo Keys - (Global) Modes Defaults ▼ Project Information Size and Quality Publish Settings Start and End ▼ Quiz Reporting Settings Pass or Fail Default Labels Styles Preview **Quiz:** ☑ Enable reporting for this project LMS: Other Standard LMSs ▾ Standard: SCORM 1.2 ▾ Configure Template: Default ▾ **Status Representation** ⦿ Incomplete ---> Complete ○ Incomplete ---> Passed/Failed **Success/Completion Criteria** ○ User Access ⦿ Slide views and /or quiz ☐ Slide Views [⦿ 100 % ○ 0 Slide(s)] ☑ Quiz Quiz is Passed ▾ Quiz is attempted **Data To Rep** ✓ Quiz is Passed Quiz Score [Quiz is passed or the quiz attempt limit is reached ☑ Interaction Data **LMS Initialization Text:** [ ] Advanced Verified LMSs: [Click here to know more] Help... OK Cancel

23) De-selecting **Interaction Data** will greatly reduce the load on the server.[3]

---

[3] "This setting affects the amount of data that gets sent to the LMS by your content. If your course module is quite large and you have lots of learners accessing it, this can place a significant load on an LMS. In fact, too much load can cause an underpowered LMS to crash. I generally deselect **Interaction Data** as this reduces load on the **LMS**. Use it only if you need it for reporting data." ***Troubleshooting Adobe Captivate versions 8 & 9***, 2015-2016 by Rod Ward, Infosemantics Pty Ltd. ALL RIGHTS RESERVED

24) **Never Send Resume Data** is not selected by default: selecting it reduces the load on the server.
 	
25) **Send Data On Every Slide** is not selected by default and can be left that way. (Captivate 9 screen)
	Preferences
	
	Category
	General Settings
	Defaults
	▼ Recording
	    Settings
	    Video Demo
	    Keys - (Global)
	    Modes
	    Defaults
	▼ Project
	    Information
	    Size and Quality
	    Publish Settings
	    Start and End
	▼ Quiz
	    Reporting
	    Settings
	    Pass or Fail
	    Default Labels
	
	Styles Preview
	
	Quiz: ☑ Enable reporting for this project
	
	LMS: Other Standard LMSs ▼
	
	Standard: SCORM 1.2 ▼ Configure
	
	Template: Default ▼
	
	Status Representation
	⦿ Incomplete ---> Complete
	◯ Incomplete ---> Passed/Failed
	
	LMS Advanced Settings
	
	LMS Advanced Settings
	    ☐ Send Data On Every Slide
	    ☑ Never Send Resume Data
	    ☑ Escape Version and Session ID
	Don't Escape Characters: [_____]
	
	Help... OK Cancel
	
	Advanced
	
	LMS Advanced Settings
	
	LMS Advanced Settings
	    ☐ Send Data On Every Slide
	    ☑ Never Send Resume Data
	    ☑ Set exit to normal after completion
	    ☐ Escape Version and Session ID
	Don't Escape Characters: [_____]
	
	Help... OK Cancel Cancel
	
	Figure 2: Captivate 8 version. Setting exit to normal just resets the slide.

28

26) The first three **Required** quiz options are self-explanatory. **NOTE:** If you select **Answer All**, it means that every question must be answered *correctly* to earn a pass.

Preferences

Category

General Settings
Defaults
▼ Recording
    Settings
    Video Demo
    Keys - (Global)
    Modes
    Defaults
▼ Project
    Information
    Size and Quality
    Publish Settings
    Start and End
▼ Quiz
    Reporting
    Settings
    Pass or Fail
    Default Labels

Styles Preview

Quiz: Settings

Quiz:

Name: Quiz

Required: Pass Required - The user must pass this quiz... ▼

    Optional - The user can skip this quiz
    Required - The user must take the quiz to continue
✔ Pass Required - The user must pass this quiz to continue
    Answer All - The user must answer every question to continue

Objective ID:
Interaction ID Prefix:
Note:

Settings:
☐ Shuffle Answers
☐ Clear
☐ Back
☐ Skip
☑ Review Mode Navigation Buttons
☐ Submit All
    Submit All Messages...
☐ Branch Aware
☑ Show Progress
    Progress Type: Relative ▼
☑ Allow Backward Movement
☐ Show Score at the End of the Quiz
    Quiz Result Messages...
☑ Allow User to Review the Quiz
    Question Review Messages...
☐ Hide Playbar in Quiz

Help...

OK   Cancel

27) Here's where eLearning designers can choose the passing criteria.
 	
28) In this case, we want the user to
• achieve a percentage of 75% or more,
• with 3 attempts.
 	
29) If the user fails after three attempts, they will continue and exit the course, but without having earned the completion credit.
	Preferences
	
	Category
	General Settings
	Defaults
	▼ Recording
	  Settings
	  Video Demo
	  Keys - (Global)
	  Modes
	  Defaults
	▼ Project
	  Information
	  Size and Quality
	  Publish Settings
	  Start and End
	▼ Quiz
	  Reporting
	  Settings
	  Pass or Fail
	  Default Labels
	
	Styles Preview
	
	Quiz: Pass or Fail
	
	Pass/Fail Options: ◉ 75 % or more of total points to pass
	○ 0 points or more to pass (Total Points: 0)
	
	Note: Maximum points to be earned is the sum of all weighted points.
	
	If Passing Grade:
	Action:
	Continue ▼
	
	If Failing Grade:
	Allow User: 3 Attempts OR ☐ Infinite Attempts
	☑ Show Retake Button
	Action:
	Continue ▼
	
	Help... OK Cancel

30) You can choose the action the lesson will perform when the user completes the quiz.

31) If the Attestation slide is the last slide in the presentation, and if you've made it clear to the learner, you can choose to exit the course when the learner clicks **Submit** to the Attestation question.

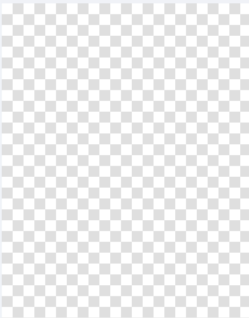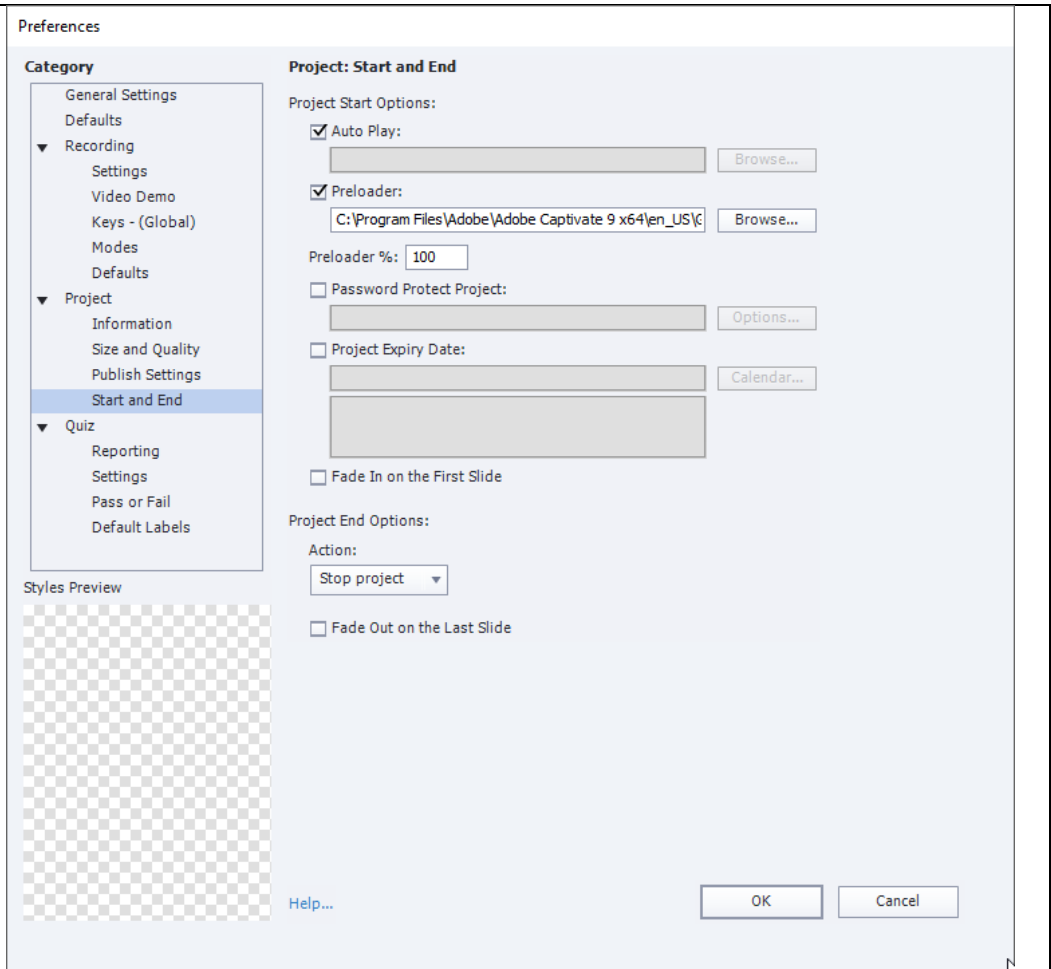|	

32) Entering the author's contact information is a best practice.	**Preferences**
	
	Category
	    General Settings
	    Defaults
	▼ Recording
	      Settings
	      Video Demo
	      Keys - (Global)
	      Modes
	      Defaults
	▼ Project
	      Information
	      Size and Quality
	      Publish Settings
	      Start and End
	▼ Quiz
	      Reporting
	      Settings
	      Pass or Fail
	      Default Labels
	
	Styles Preview
	
	Project: Information
	
	Author:
	Company:
	E-mail: author@company.com
	Website: www.company.com
	Copyright:
	Project Name: This is the Project Name on Project: Information page.
	Description: This is the Project: Information Description.
	
	Time: 15.0secs (450 frames)
	Resolution: 1024 x 627
	Slides: 2
	Hidden Slides: 1
	
	Help... OK Cancel

33) The **Project Name** that you enter on the **Project Information** screen appears in the browser header when the eLearning lesson is played.	
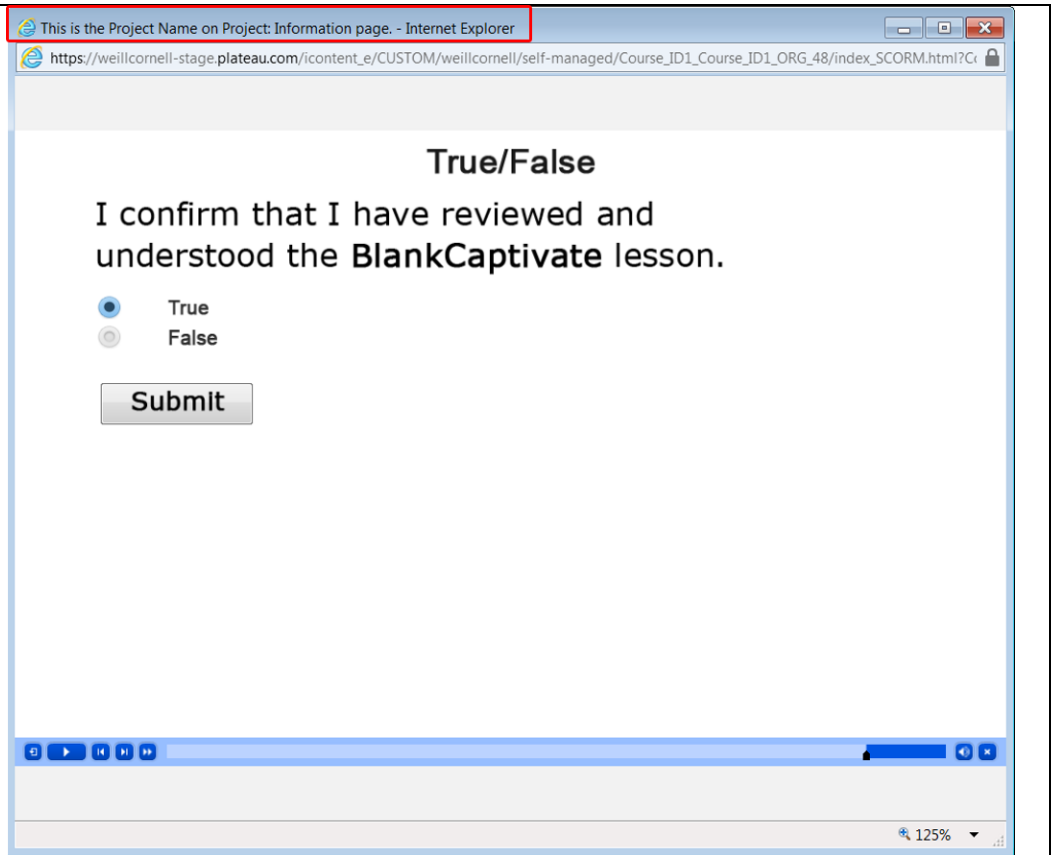
34) The best practice, when uploading content to an LMS, is to use **Stop Project**. Do not use **Close Project**, for the reasons stated below.[4]

Appendix 1 SCORM 1.2 vs. SCORM 2004

(Source of Information and citations: https://www.paradisosolutions.com/blog/scorm-1point2-vs-scorm-2004/.)

Currently, SuccessFactors supports SCORM 1.2 (all editions) and SCORM 2004, 2nd and 4th editions.

First major difference: Single-Status Reporting vs. Split-Status Reporting

SCORM 1.2 has one value to hold the status of the lesson – 'lesson_status'. This lesson status can be

- passed,
- failed,
- completed,
- incomplete,
- browsed, or
- not attempted.

So, wth a Captivate eLearning lesson, all you can do is have the lesson transmit one of these statuses to your LMS, which will give your learner one of these 6 "marks" or "grades".

From Captivate you can set your lesson up to indicate one of these statuses (only one) either by having the user …

- access the course (simply opening it), or …
- view a certain percentage (or all) of the slides, or …
- pass a quiz according to the passing standard you've established in the quiz settings … or
- view some or all of the slides <u>and</u> passing the quiz.

Suppose you, the instructor, want to know if the learner has completed the lesson ("lesson completed"), but not passed the quiz ("quiz failed"). You can't do that with SCORM 1.2, but you can with SCORM 2004.

SCORM 2004 can provide two results, answer two questions, not just one.

- Has the user completed the course? (yes/no, according to the standards you've established)
- Has the user passed the course? (yes/no, according to the standards you've established)
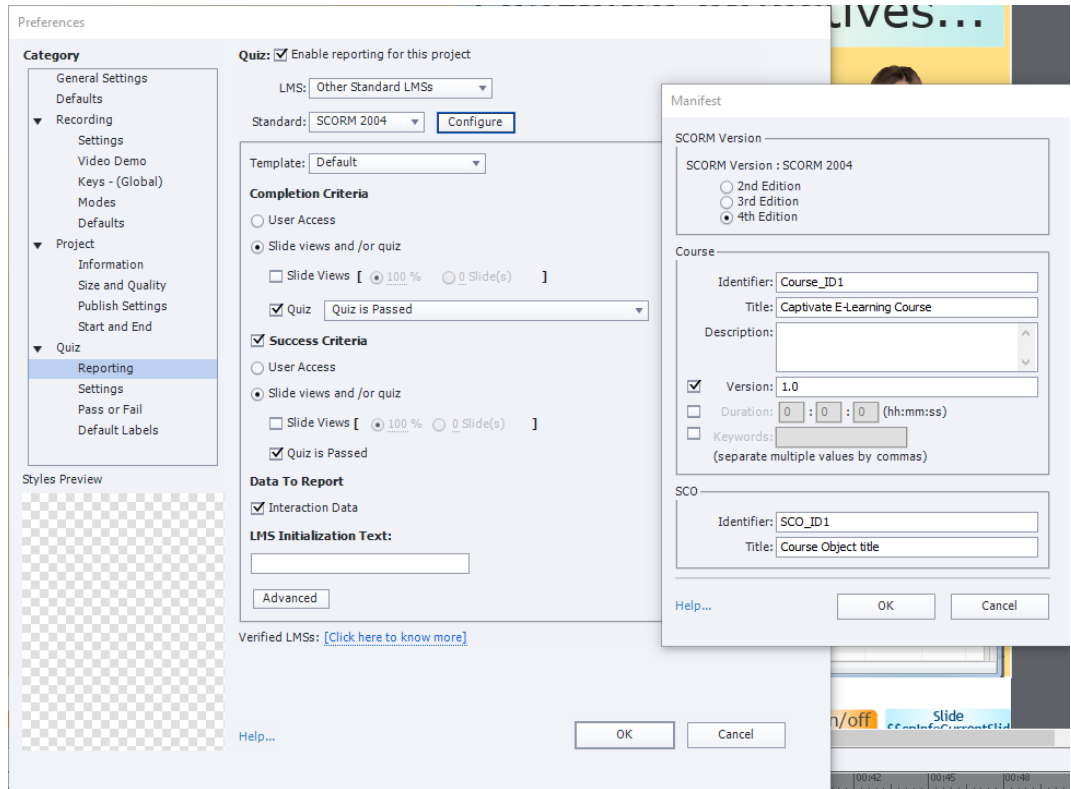
Note: This could be useful if you have a pre-test where, if the user passed, he or she didn't have to complete the course. Statuses would be: Course passed // Course incomplete. If you knew that your users already knew the material, then you could eliminate the requirement to take the entire course.

Is this so-called "split-status reporting" needed? Take into consideration the fact that Items in SuccessFactors currently just have "Complete" or "Incomplete" status, based on whether the user has completed the various content objects and quizzes. It's not clear how this feature would be implemented.

Second Major Difference: Read/Write Interactions

"SCORM 2004 specifies interactions to be read/write, which is especially helpful not only for reporting but your lesson can now query the status of a previous interaction, get the result, and act accordingly (i.e. 'user answered this question in the last launch, so they don't get another chance to answer it again')."

Using this feature involves configuring the LMS. SuccessFactors may not support these kinds of interactions.

Third Major Difference: Sequencing

"SCORM 1.2 also lacked a sequencing and navigation specification that allowed the content vendor to specify how the learner was allowed to progress between SCOs. The lack of a sequencing specification meant that most SCORM 1.2 content was produced as a single monolithic SCO instead of created with granular, reusable SCOs. SCORM 2004 addressed both of these problems."

SuccessFactors LMS enables "forced sequential completion" within an item among Sharable Content Objects (SCOs), so this feature may not be needed.

## Appendix 2: Eliminating non HTML5-Compliant Elements

NOTE: This is really for when you want your eLearning to be totally HTML5 compliant so that it can be accessed on tablets, smartphones, and other non-mouse devices.

Eliminate all non HTML5-compliant components (Rod Ward)

Building for **HTML5** means your course cannot contain any component that would not work on a mobile device browser. This is the part most Captivate developers think they can easily handle, but you may be surprised to know the list of potential non-compliant components is quite long, and they're not all *obvious* candidates for removal.

Remove all Flash SWF animations and SWF graphics

You will have known by now that all animations created in **Flash** and imported into the Captivate project as **SWF** need to be removed. But it has also become a common (though not recommended) practice to capture software simulations in Captivate, publish them out as **SWF** files, and then insert these into another **CPTX** project file. All **SWFs** have to go, regardless of where they came from.
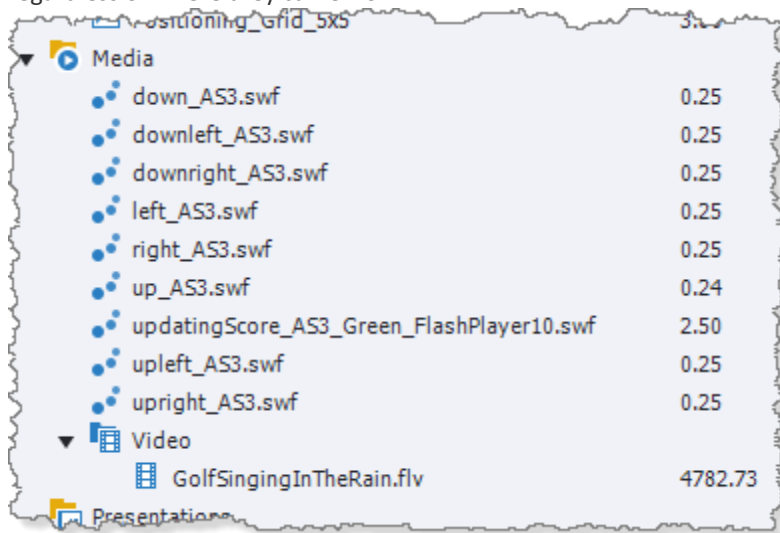
To discover at least *some* of the **SWF** components that infest your project file, you can start by looking in the **Project Library** (shown in the screenshot at right).

For example, note that the **Media** section of this **Library** lists quite a number of **SWF** files used in the project. The **Video** section below it also contains an **FLV** file. (More about **FLV** below.)

SWFs aren't necessarily *animations*. Some graphics editors allow you to save a *static* graphic as **SWF** format. Though the original **PNG** or **JPG** file would be HTML-compliant, the **SWF** version of the same file would not be.

No matter whether the **SWFs** were originally created in **Adobe Flash**, **Adobe Captivate**, or some other application you need to replace them, because they won't work in mobile browsers due to the fact that none of them have a **Flash Player** plugin.

Remove PowerPoint-based slides

This one is going to hurt big-time if you happen to be one of those e-learning developers accustomed to building their course content in **PowerPoint** and then importing that **PPT** deck into Captivate. The fact is that Captivate converts these slides into...you guessed it...**SWF** animations.

The screenshot at right shows the **Properties** data for a slide in a Captivate project created directly from a **PPT** deck. Note that the background of this slide is shown to be an **SWF** file. (If the original **PPT** slide contained animations, the **SWF** background for this slide would also be animated.)

The fact that **PPT** slides become **SWFs** is the reason Captivate disables the option under **File > Import > from PowerPoint slides** in any responsive project. Yes, that option still exists in *normal* projects to allow publishing to **SWF**. But when you publish out to **HTML5**, the background **SWFs** are converted to static **PNG** files instead.

So unless you are happy for all of your **HTML5** courses to consist of little more than frozen background images, you need to forget about using **PowerPoint** from now on and build everything from the ground up in Captivate, using only supported slide objects.

Change SWF Pre-loaders to Animated GIFs

Did you know that Captivate's own pre-loaders (specified in **Preferences > Project > Start and End > Preloader**) are actually **SWF** files? Note the default.**swf preloader** file in the screenshot shown below from a project that was originally created in Captivate 6.
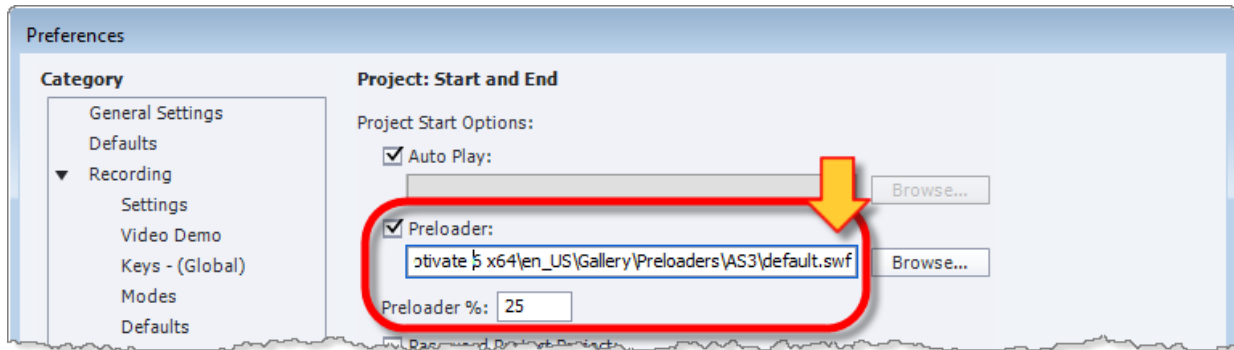
Captivate *does* offer a selection of **animated GIFs** as alternative pre-loader animations, but be aware that these don't offer all the same functionality as the **SWF** pre-loaders. They mainly give you something to look at while the project loads up.

Change SWF project skins to HTML5 skins

Project skins from older versions of Captivate were also usually **SWF**. To play safe, reset the skin to one of the defaults provided in the latest Captivate version that can publish to either format. That skin is more likely to have an **HTML5** equivalent.

Remove or convert all FLV Videos

In the days before Captivate had **Demo** and the **CPVC** capture format, full-motion video captures in software simulations were saved as **Motion Video** or **FMV** format. This format is also not going to work in **HTML5**. You will either need to remove these components or else convert them to a more compliant format such as **MP4**.

Remove all AS2 or AS3 Widgets

Since Captivate 4 it has also been possible for **ActionScript** programmers to build **SWF** widgets insert these into projects to enhance extend functionality. In very early examples, these may have been built with **ActionScript 2**, but since Captivate 5 arrived, only **ActionScript 3** widgets could be inserted. Either way, all such **SWF** widgets must be removed. (Don't bother even looking for **HTML5** equivalents for these widgets. Most widget developers gave it up as a bad idea.)

While on the subject of widgets, you may not have known that Captivate's **Text Animation** objects are also **SWF** widgets. So you need to remove them or replace them with static text captions instead.

Captivate's **Learning Interactions** are also widgets, but those that ship with the latest versions of Captivate should be **HTML5-compliant.** If in doubt, replace them with current versions.

Remove rollover objects

In the mobile world, where there are no computer mice and everything is done with your finger or a stylus, *rollover* and *rollout* events become redundant. And that means Captivate objects such as **Rollover Captions**, **Rollover Images** and **Rollover Slidelets** also have no place. Remove them or replace them with **Advanced Actions** based on click or tap events.

Remove slide transitions

And one last thing...you need to remove all slide transitions. Those nice fades or wipes effects when moving from one slide to the next are all based on (you guessed it) **SWF** animations, and are therefore not compliant with **HTML5** output.

Using the HTML5 Tracker

To be fair, Adobe saw this issue coming several versions ago and kindly included a tool called the **HTML5 Tracker** that is designed to show Captivate developers all objects or elements in their project that would be unsupported in **HTML5** output.
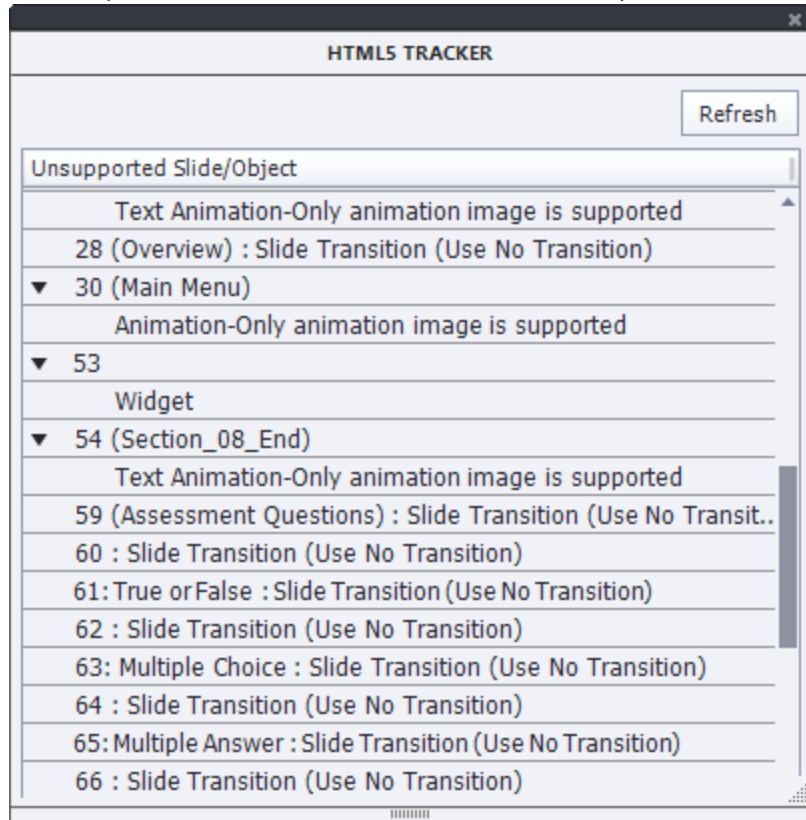
For example, in the screenshot at right you can see that it lists unsupported objects under the slides on which they can be found. Scanning the list you will see many of the objects we have discussed above, including: **Text Animations**, **Widgets**, and **Slide Transitions**. But the tool isn't perfect.

You cannot trust the **HTML5 Tracker** to find every single issue for you. And unfortunately it has sometimes given 'false positives', reporting something as unsupported when in fact it is. (For example, the **HTML5 Tracker** in Captivate 8/9 reports several supported **Effects** as unsupported.) All things considered though, it's still a great place to start.