

Content Development Considerations for SF Learning

This guide is intended for content developers creating courseware for the SuccessFactors Learning LMS (SF Learning). The guide covers the specifics of the AICC and SCORM content runtime environments as implemented by SuccessFactors, and details the most common issues encountered when integrating courseware with the SF Learning LMS.

Supported E-learning Standards

AICC HACP

Supported Data Elements: AICC		
Element	Read	Write
[CORE]		
STUDENT_ID	◇	
STUDENT_NAME	◇	
SCORE (min, max, raw)	◇	◇
TIME	◇	◇
CREDIT	◇	
LESSON_LOCATION	◇	◇
LESSON_STATUS	◇	◇
[Core_Lesson]	◇	◇
[Objectives_Status]		
J_ID.n	◇	◇
J_Score.n	◇	◇
J_Status.n	◇	◇
[Core_Vendor]	◇	

SCORM 1.2

Supported Data Elements: SCORM 1.2		
Element	Read	Write
cmi.core._children	◇	
cmi.core.student_id	◇	
cmi.core.student_name	◇	
cmi.core.lesson_location	◇	◇
cmi.core.credit	◇	
cmi.core.lesson_status	◇	◇

Supported Data Elements: SCORM 1.2 (Continued)		
Element	Read	Write
cmi.core.entry	◇	
cmi.core.score._children	◇	
cmi.core.score.raw	◇	◇
cmi.core.score.max	◇	◇
cmi.core.score.min	◇	◇
cmi.core.total_time	◇	◇
cmi.core.lesson_mode	◇	
cmi.suspend_data	◇	◇
cmi.launch_data	◇	
cmi.objectives._children	◇	
cmi.objectives._count	◇	
cmi.objectives.n.id	◇	◇
cmi.objectives.n.score._children	◇	
cmi.objectives.n.score.raw	◇	◇
cmi.objectives.n.score.min	◇	◇
cmi.objectives.n.score.max	◇	◇
cmi.objectives.n.status	◇	◇
cmi.core.exit		◇
cmi.core.session_time		◇

SCORM 2004 2nd and 4th Editions

SCORM 2004 3rd Edition courseware will launch under the SCORM 2004 2nd Edition runtime in the SF Learning environment. During the content import, the course administrator will receive a warning that the SCORM 2004 3rd Edition runtime is not officially supported by SF Learning.

Supported Data Elements: SCORM 2004 (2nd & 4th Editions)		
Element	Read	Write
cmi._version	◇	
cmi.learner_id	◇	
cmi.learner_name	◇	
cmi.location	◇	◇
cmi.credit	◇	
cmi.completion_status	◇	◇
cmi.entry	◇	
cmi.exit		◇
cmi.progress_measure	◇	◇

Supported Data Elements: SCORM 2004 (Continued)		
(2nd & 4th Editions)		
Element	Read	Write
cmi.completion_threshold	◇	
cmi.success_status	◇	◇
cmi.score._children	◇	
cmi.score.raw	◇	◇
cmi.score.max	◇	◇
cmi.score.min	◇	◇
cmi.score.scaled	◇	◇
cmi.scaled_passing_score	◇	
cmi.total_time	◇	
cmi.session_time		◇
cmi.mode	◇	
cmi.max_time_allowed	◇	
cmi.time_limit_action	◇	
cmi.suspend_data	◇	◇
cmi.launch_data	◇	
cmi.comments_from_learner._count	◇	
cmi.comments_from_learner._children	◇	
cmi.comments_from_learner.n.comment	◇	◇
cmi.comments_from_learner._location	◇	◇
cmi.comments_from_learner.n.timestamp	◇	◇
cmi.comments_from_lms.n.comment	◇	
cmi.comments_from_lms._children	◇	
cmi.comments_from_lms._count	◇	
cmi.comments_from_lms.n.location	◇	
cmi.comments_from_lms.n.timestamp	◇	
cmi.objectives._children	◇	
cmi.objectives._count	◇	
cmi.objectives.n.id	◇	◇
cmi.objectives.n.score_children	◇	
cmi.objectives.n.score.scaled	◇	◇
cmi.objectives.n.score.raw	◇	◇
cmi.objectives.n.score.max	◇	◇
cmi.objectives.n.score.min	◇	◇
cmi.objectives.n.progress_measure	◇	◇
cmi.objectives.n.success_status	◇	◇
cmi.objectives.n.completion_status	◇	◇
cmi.objectives.n.description	◇	◇
cmi.learner_preference._children	◇	
cmi.learner_preference.audio_level	◇	◇
cmi.learner_preference.language	◇	◇
cmi.learner_preference.audio_captioning	◇	◇

Supported Data Elements: SCORM 2004 (Continued) (2nd & 4th Editions)		
Element	Read	Write
cmi.learner_preference.delivery_speed	◇	◇
cmi.interactions._children	◇	
cmi.interactions._count	◇	
cmi.interactions.n.id	◇	◇
cmi.interactions.n.type	◇	◇
cmi.interactions.n.weighting	◇	◇
cmi.interactions.n.learner_response	◇	◇
cmi.interactions.n.result	◇	◇
cmi.interactions.n.latency	◇	◇
cmi.interactions.n.description	◇	◇
cmi.interactions.n.timestamp	◇	◇
cmi.interactions.n.objectives._count	◇	
cmi.interactions.n.objectives.n.id	◇	◇
cmi.interactions.n.correct_response._count	◇	
cmi.interactions.n.correct_response.n.pattern	◇	◇
adl.nav.request	◇	◇
adl.nav.request_valid.continue	◇	
adl.nav.request_valid.previous	◇	
adl.nav.request_valid.choice.{target=STRING}	◇	

IMPORTANT NOTE: The SF Learning LMS does not support cmi.interactions data for AICC or SCORM 1.2 content.

Content Launch

By default content launches from SF Learning into a new browser window. Customers also have the option to configure content to launch into an inline frame of the LMS application window.

The way a course launches from SF Learning may impact design and development decisions.

Formatting & Design Considerations When Launching to a New Browser Window

The default behavior for the SF Learning LMS is to launch courseware into a new browser window.

AICC & SCORM 1.2

- Size and Formatting of the Content Window: The SF Learning Admin can configure the default size and display options for the content window. These settings are globally defined individually for AICC and SCORM 1.2 content launches.
 - Default Settings: width=800, height=600, top=0, left=0, toolbar=0, location=0, directories=0, status=1, menubar=0, scrollbars=1, resizable=1
 - Courseware may resize the content window with the `resizeTo(w,h)` javascript method.

- **Exiting Content:** Responsibility for closing the course window falls to the content or the end user. It is recommended that the course include an exit button as part of the course. The same course functionality should be triggered when using either the exit button in the course or the close button in the browser.

SCORM 2004

- SCORM 2004 content launches inside of an LMS-provided navigation wrapper. A navigation banner (82 pixels in height) displays across the top of the content window. The course content is displayed in a frame below the navigation banner. The SF Learning SCORM 2004 course wrapper does not include a course structure frame.
- The formatting of the SCORM 2004 content window is not configurable in SF Learning.
 - The SCORM 2004 window includes the browser toolbar, statusbar, menubar, auto scrollbars, and is resizable.
 - Courseware may resize the SCORM 2004 content window with the `parent.resizeTo(w,h)` javascript method.
 - An opener window in the courseware may be employed to launch the content into a new window with the desired formatting.
- **Exiting Content:** Navigation requests to load a new sco or to exit the content may be made from the lms-provided navigation banner. The courseware may also set navigation requests via API calls which are processed when the course calls `Terminate`.

Formatting & Design Considerations When Launching to an Inline Frame in SF Learning

Individual courses can be configured in SF Learning to launch in an inline frame under the SF Learning application. The size of the content frame is globally configurable as a relative percentage of the SF Learning application window. The default setting will open the content into an `iframe` that is 80% of the width and height of the SF Learning window. Confer with your customer on their settings. Using a responsive design approach for your courseware is recommended.

Finding the SCORM API

The SCORM API is implemented in the application window when the content is launched. The relative location of the API to the content will vary depending on the content launch option and the version of SCORM used.

IMPORTANT NOTE: The SF Learning application is accessed as part of a larger HCM platform which is hosted under a different domain than the learning application. To avoid cross-site scripting errors, content should start looking for the API at the lowest level of the window hierarchy and work outwards through parent frames.

SCORM 1.2

- **Launch in a new window:** Content will find the API in its opener window. (`window.opener`).
- **Launch in an inline frame:** Content will find the API in its parent window. (`window.parent`).

SCORM 2004

SF Learning launches SCORM 2004 content into a frameset containing the SCORM 2004 navigation wrapper. This wrapper includes a local pointer object tied to the API in the SF Learning window.

- **Launch in a new window:** Content will find the API in both its parent window (`window.parent`) and the wrapper's opener window (`window.top.opener`).
- **Launch in an inline frame:** Content will find the API in its parent window (`window.parent`).

Tracking Course Completion and Failure

AICC & SCORM 1.2

Completion is driven by the lesson_status value received from the content. A mastery score may also contribute in determining completion.

SF Learning will grant credit for content *without a mastery score* when:

- lesson_status value of completed OR passed is received.

SF Learning will grant credit for content *with a mastery score* when:

- lesson_status value of passed is received and NO raw score value is sent.
OR
- lesson_status value of passed, completed, or failed AND a raw score value that is greater than, or equal to, the mastery score are received.

SCORM 2004

Each SCO and aggregation in a SCORM 2004 content package has a primary objective. SF Learning will grant credit when the status of the primary objective is 'satisfied'.

Below are the two most common methods by which SCORM 2004 content can set a SCO's primary objective to 'satisfied' and trigger completion in SF Learning.

- The content sends a cmi.completion_status value of completed AND a cmi.success_status value of passed.
- If the SCO's primary objective has a satisfiedByMeasure value of true and a minNormalizedMeasure value specified, the primary objective will be satisfied when SF Learning receives a cmi.score.scaled value that is greater than, or equal to, the minNormalizedMeasure value.

```
<imsss:objectives>  
  <imsss:primaryObjective objectiveID="PRIMARYOBJ" satisfiedByMeasure="true">  
    <imsss:minNormalizedMeasure>0.8</imsss:minNormalizedMeasure>  
  </imsss:primaryObjective>  
</imsss:objectives>
```

In addition, SCORM 2004 content may use advanced sequencing rules, local or global objectives, and rollup rules in the manifest to trigger completion in SF Learning.

IMPORTANT NOTES:

- In SF Learning, the mastery score value set in the LMS is not made available to the content at runtime.
- The content should wait until the user has viewed all the content and answered all the required questions before sending data that would trigger a completion in the LMS.

Tracking Failed Attempts

SF learning allows administrators to configure courses to track a learner's failed attempts. In order for the failed attempts to be recorded correctly:

- The user must exit the content at the end of each attempt
- The content should wait until the user has viewed all the content and answered all the required questions before sending the lesson_status or success_status of failed.

Additional Considerations

SCORM 2004: cmi.exit and Data Persistence

In order to ensure the content runtime data is maintained for subsequent attempts, the content must set cmi.exit to 'suspend' prior to calling Terminate.

Exit Flag - logout

When an AICC or SCORM 1.2 course sends an exit value of 'logout', the user will be logged out of the SF Learning application. To prevent users from being logged out of the learning application, this value should be set to 'suspend' or to an empty string,

Tracking Objective Data

AICC and SCORM 1.2 objective data will not be tracked unless an objective record exists and is associated to the content object (module) in SF Learning.

- AICC: SF Learning does not support the AICC .ort file for import. Objective records need to be created and associated to the content object manually.
- SCORM 1.2: SF Learning supports the importing objectives from a SCORM 1.2 manifest. The objective ID is pulled from the <description> element under a <classification> category where the <purpose> element for the <classification> contains a <value> of "Educational Objective".

```
<classification>
  <purpose>
    <source>
      <langstring xml:lang="x-none">LOMv1.0</langstring>
    </source>
    <value>
      <langstring xml:lang="x-none">Educational Objective</langstring>
    </value>
  </purpose>
  <description>
    <langstring>Obj_1</langstring>
  </description>
</classification>
```

SCORM 1.2 & 2004: Populating Content Object Title & Description

When SCORM content is imported into SF Learning, the title and description text for the resulting module (content object) record(s) are pulled from the Learning Object Metadata (LOM) associated to the resource in the imsmanifest.xml file.

```
<lom>
  <general>
    <title> <string language="en-us">Conduct of Vessels in any Condition of
      Visibilty</string></title>
    <description>
      <string language="en-us">
        Discusses general rules of operation for vessels on inland waters.
        Topics discussed include: Lookout, Safe Speed, Collision, Channels
        and Traffic Separation.
      </string>
    </description>
  </general>
</lom>
```

Best Practices When Developing for the SCORM Runtime Environment (RTE)

The following recommendations, taken from the ADL SCORM 2004 RTE documentation, offer ways to minimize potential data communication issues during course runtime. These recommendations also apply to SCORM 1.2 and the equivalent API methods.

- Call Commit("") after any significant event occurs in the courseware that must but recorded.
- Do not call Commit("") after every SetValue() call as this may affect courseware performance. Call Commit("") after a reasonable group of SetValue() calls are made.
- Call SetValue() and Commit("") throughout the runtime of a course rather than waiting to send a large amount of data just prior to calling Terminate("").
- Call Terminate("") before the courseware is unloaded. If supported by the client's browser, Terminate("") should be called in an onbeforeunload event. For browsers that rely only on the onunload event, it would be advisable to provide an alternative mechanism to trigger termination of the content session, i.e., an internal course exit button. Courseware may be coded to trigger the Terminate("") call through multiple events, however, it should track when the call is successfully executed and not invoke the call again.

In addition to the recommendations above, courseware should employ the GetLastError() method to confirm API calls are successful. The courseware should alert the user to any problems and, where possible, provide potential steps for resolution.

SCORM 2004 Sequencing & Navigation Settings

SCORM 2004: Hiding Navigation Buttons

SF Learning launches SCORM 2004 content inside an LMS-provided navigation wrapper.

If your content is coded to set the required navigations requests and the LMS-provided navigation buttons are not needed, they can be hidden by adding the prescribed tags below in the imsmanifest file.

- SCORM 2004 2nd Edition – Previous, Continue and Exit can be hidden.

```
<adlnav:presentation>
  <adlnav:navigationInterface>
    <adlnav:hideLMSUI>previous</adlnav:hideLMSUI>
    <adlnav:hideLMSUI>continue</adlnav:hideLMSUI>
    <adlnav:hideLMSUI>exit</adlnav:hideLMSUI>
  </adlnav:navigationInterface>
</adlnav:presentation>
```

- SCORM 2004 4th Edition – Previous, Continue, Exit, and Suspend can be hidden

```
<adlnav:presentation>
  <adlnav:navigationInterface>
    <adlnav:hideLMSUI>previous</adlnav:hideLMSUI>
    <adlnav:hideLMSUI>continue</adlnav:hideLMSUI>
    <adlnav:hideLMSUI>exitAll</adlnav:hideLMSUI>
    <adlnav:hideLMSUI>suspendAll</adlnav:hideLMSUI>
  </adlnav:navigationInterface>
</adlnav:presentation>
```

DeliveryControls: completionSetByContent & objectiveSetByContent

Under the SCORM 2004 standard, the LMS sequencing engine will mark a sco complete if completionSetByContent and objectiveSetByContent are "false" (default value for both), and the sco is exited without setting a completion_status and success_status value. To avoid unearned completions, please set both completionSetByContent and objectiveSetByContent to "true" for all scos.

```
<imsss:deliveryControls completionSetByContent="true" objectiveSetByContent="true"/>
```

controlMode: choice

- If set to 'true' (default value), the learner is allowed to launch any sco within the course structure.
- If set to 'false', the learner is required to view the content in the order defined in the imsmanifest.
 - In the SF Learning SCORM 2004 2nd Edition runtime: If the learner exits the course with the "Exit" navigation button, SF Learning will remove the learner's access to resume the course.

```
<imsss:controlMode choice="true" flow="true"/>
```

controlMode: flow

If set to 'false' (default value), the learner is NOT allowed to navigate between scos using the 'Continue' and 'Previous' buttons in the LMS-provided navigation banner.

```
<imsss:controlMode choice="true" flow="true"/>
```