Custom Content in Impression

Logicdriven, LLC

Contents

Overview

This document describes the custom data capabilities in the Impression Learning Content Framework.

Introduction

The Impression product has always had a uniquely adaptable design, combining the flexibility of a custom solution with the efficiencies of an off-the-shelf product. The Runtime Engine Developer's Kits allow you to take the courseware data developed in the Content Creation Tool and present it in any fashion desired. Because each runtime is built from source code, the only limitations on capabilities are those imposed by the development toolset and deployment environment; whether you use Adobe Flash, HTML/JavaScript, Unity3D, or something else.

But a custom runtime is only part of the solution. A runtime can only work with the data it is provided with, and (assuming that no other tools are used to create content fed to a runtime) that means that the CCT needs to be flexible, too.

With the configuration capabilities of the CCT, there are a number of basic options you can set for each storyboard type—for example, you can choose whether to support four or five question/answer pairs for matching questions. You can also specify options common to all storyboards, like a narration audio file, student directions, or remediation types for questions.

Some storyboards types support "open-ended" data fields. For example, the Canvas storyboard supports an action type of "Command". When this action type is selected for an element, the content creator specifies the exact command for the element by entering a text value in the Command Text field. The runtime framework has no intrinsic support for any specific Command action types; the runtime engine developer must add support for the command to the player. Other examples of open-ended fields include the Placeholder Data field on the Placeholder storyboard type, and the common Configuration and User Defined fields. In fact, most of the fields on the General Tab are open-ended—it's the responsibility of the runtime to determine what to do with the (as one example) Student Directions and Alternate Directions fields.

Looked at from a certain point of view, almost every field can be considered as an open-ended field. If you want to use the Distractor 3 field on a multiple-choice question to store the screen style, a list of related objectives, and the initials of the reviewer who approved the question; well, you can do it, although your content creators may have a hard time keeping things straight.

Fortunately, there's no need to take things to a ridiculous extreme. The CCT allows you to define additional fields and customized storyboard types so that data needed by your runtime can be entered with a minimum of hassle.

Custom Fields and Field Collections

Every field in a storyboard has some defined mechanism for editing the field. A Multiple-Choice Text question provides multiline text boxes to allow the developer to edit the answer and distractor values; a Multiple-Choice Graphics question uses asset selectors to let the developer pick the graphic from the media folder. Custom fields also need a mechanism for editing—the table below lists the possible options:

Editor Type Name	Value Type	Editor Description
Single line text (spellcheck)	Text	A single line textbox with spellcheck enabled.
Single line text (no spellcheck)	Text	A single line textbox without spellcheck.
Multiline Text (spellcheck)	Text	A multiline textbox with spellcheck enabled.
Multiline Text (no spellcheck)	Text	A multiline textbox without spellcheck.
List items (single-select)	Text	A drop-down combobox. Users may select any one value. Values are specified along with the field definition.
List items (multiselect)	Text	A list of values. Users place a checkmark next to the values they want to select. Possible values are specified along with the field definition.
Graphic	AssetData	An asset selection widget. Clicking on the ellipsis button to the right of the filename brings up the media selector window with filters for still graphics.
Video	AssetData	An asset selection widget. The media selector window uses filters for videos.
Sound	AssetData	An asset selection widget. The media selector window uses filters for sounds.
Flash File	AssetData	An asset selection widget. The media selector window uses filters for Flash files.
Asset (generic)	AssetData	An asset selection widget. The media selector window uses filters for general assets.
Boolean (yes/no)	Boolean	A checkbox. If checked, the value is true, if unchecked, the value is false.
Number	Number	A small, single-line textbox limited to numeric input.
Offset (x/y)	Rectangle	A pair of numeric input textboxes labelled “x:” and “y:”.
Position (left/top)	Rectangle	A pair of numeric input textboxes labelled “left:” and “top:”.
Size (width/height)	Rectangle	A pair of numeric input textboxes labelled “width:” and “height:”.
Dimensions (left/top/width/height)	Rectangle	Four numeric input textboxes labelled appropriately.
Coordinate Pair (x1/y1/x2/y2)	Rectangle	As with Dimensions, but with different labels.
Color	ColorData	A box showing the selected color and its RGB value. An ellipsis button on the right edge displays a standard Windows color picker when clicked.
Font	FontData	A box showing the selected font name and size. An ellipsis button on the right edge displays a standard Windows font picker when clicked.
Storyboard Location	LocationData	A single box showing the title of the selected storyboard. An ellipsis button on the right

Editor Type Name	Value Type	Editor Description
		edge displays a dialog allowing users to select a storyboard.
Storyboard Range	LocationData	A pair of boxes showing the titles of the beginning and end of a range of storyboards. The ellipsis button on the right edge of each box shows a dialog allowing users to select a storyboard.

The CCT does not support adding custom fields directly to a storyboard; instead, an ordered collection of fields is defined and the entire collection is presented for editing.

In addition to the editing widgets listed above, a number of additional non-field options can be added to a collection to enhance the content development experience. These are listed in the following table:

Option Name	Description
(space)	Adds several pixels of spacing between elements. Often used between groups of fields.
(section break)	A horizontal line. Used to separate sections.
(element instructions – below element)	A text description shown directly under the preceding element.
(instructions)	A text description with some spacing shown left justified.
(section title)	A text description shown in a larger font than the instructions or element instructions options.

Using Custom Fields in the CCT

Custom field collections are supported with the following items.

The Command Storyboard Type

The Command storyboard type allows the content developer to choose one of any number of defined collections (called *commands*). The available commands are defined in the type-specific properties for the Command storyboard.

Each command can also specify whether or not it should be treated as a question type. If the properties specify that it should be a question, the question-related properties on the General tab will be visible and editable. You can also specify that the storyboard may or may not be a question; if this is specified, the CCT will look to see if a Boolean property named “IsQuestion” is present and has a value of true.

The Sequencer Storyboard Types

The Command Sequencer storyboard type includes both a single set of common fields (defined per project) as well as an ordered list of commands. Any number of commands can be defined; the sequencer allows the content developer to create and order as many commands as needed.

Both the common fields and the Command Sequencer commands are defined in the type-specific properties for the Command Sequencer storyboard. Like the Command storyboard, the Command Sequencer storyboard type can be defined as being a question.

Two variations of the Command Sequencer are available; the Audio Sequencer and the Time Sequencer. Both of these storyboard types support a common field collection and any number of commands. The Audio Sequencer also supports a single common audio file, and each command includes a time offset indicating (typically) the point in the audio file at which the command should be processed. The Time Sequencer also includes a time offset in each command, but does not have an associated asset.

Storyboard Commands

In addition to the storyboard types listed above, a collection of fields can be added to any existing storyboard (including the Command and sequencer storyboards). These collections are called Storyboard Commands, and are defined from the main tab of the Project Properties dialog.

Any number of storyboard commands can be defined and created.

Runtime Framework Support

For the most part, there is no support within the Base Class Framework for custom fields. Although the BCF includes players for the Command/Audio (AS3 BCF only)/Time Sequencer storyboard types, support is limited to collating the data and playback-related methods and events for the Audio and Time sequencers. There is no explicit support for the Command storyboard or for Storyboard Commands.

Custom Field Examples

The AS3 and HTML/JavaScript MyShell sample runtimes include examples of custom fields and storyboard types:

- The Storyboard Command feature is used to add Warnings, Cautions, and Notes to any storyboard.
- The Command Sequencer storyboard type is used to present a list of items. When each item is clicked, additional text and a supporting graphic is revealed.
- The AS3 MyShell uses the Audio Sequencer and Time Sequencer storyboard types to progressively reveal text and images.
- The HTML5/JavaScript MyShell uses the Command storyboard to define Pretest and Posttest question collections, add support for the Multiple-Select storyboard type, and to redefine the Multiple-Choice Text storyboard type to include additional distractors.

We’ve also seen or used the custom field capabilities to:

- Hold information for custom reports (Instructor Guide/Student Guide).
- Store commenting and approval information during the production process.
- Specify popup text and/or graphics for specific phrases found in the storyboard text.
- Store detailed objective and reference data.
- Transform a standard Canvas storyboard into a "Drag and Drop to Hotspot" question.
- Replace standard question types to include additional capabilities or remediation options.
- Specify callout or other specially formatted text content.
- Create any number of custom storyboard types.

Multiple-Select Example

The runtime Base Class Framework includes support for the "Multiple-Select" storyboard type. Often called "Multiple Choice/Multiple Select", the student is presented with a question and must choose one or more responses as an answer.

This storyboard type has no editor built into the CCT. This was a deliberate decision intended to provide a concrete example of using the Command storyboard type to support a new capability. A technical article available on the Impression web site includes a step-by-step walkthrough for adding a multiple select Command storyboard to a content database.

Conclusion

The Impression Content Creation Tool includes a number of mechanisms that can be used to specify additional courseware data. These include "open-ended" fields, new storyboard types, and Storyboard Commands that can be added to existing content.

These capabilities help the content side of the tool approach the flexibility and power of the runtime DevKits while keeping the courseware editing experience simple and focused.