InEventScript 1.0.3

Overview

InEventScript is a plug-in for use with Adobe® InDesign® CS or CS2. It allows you to start scripts at predefined moments during the use of InDesign. For installation instructions, please refer to the **ReadMe.txt** file found in this file package. This document details features of the user interface within InDesign, manual configuration of the InEventScriptData.xml file and product registration.

InEventScript Palette

The InEventScript palette shows the current event to script associations and displays a hierarchical list of InDesign's menus and actions available for association. Between the association list and the hierarchical list is a row of buttons, one on the left and two on the right. The button on the left shows and hides the hierarchical view. The buttons on the right change or remove, respectively, the selected association.
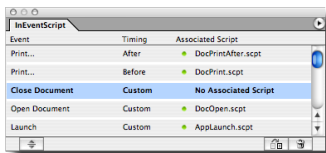

Figure 1 InEventScript Association List

InEventScript Association List

The event to script association list appears at the top of the InEventScript palette. It shows the name of the event, the

timing of the associated script, a script state icon and the script name/location.

InDesign menu events appear in the palette with identical names as shown in the menu items for the application. This means that for some context sensitive menu events the actual text may change during the normal use of InDesign.

Script Timing

The Timing field typically reads "Before" or "After", indicating whether the script runs before or after the given event. In the case of custom events the timing reads "Custom", as the timing is specific to that event. See Custom Events later in this document for more information.

Script State

There are three possible states for the script availability icon, which appears immediately to the left of the script name in the Associated Script field. A green dot (●) indicates the script file exists at the given location. A warning icon (⚠) means the script file does not exist at the chosen location. No icon (blank) indicates the event does not have an associated script.

Script Location

The Associated Script field shows the filename of the script associated with this action and timing. Hovering the cursor over the script file name will display a tool-tip window containing the entire pathname for the script relative to the InDesign application.

Table 1 describes the functionality available to the user via the top list. To associate a different script with an event

2

the user can double-click on the list item and use the resulting file chooser dialog. Additionally, they may Alt-click (Windows) or Option-click (Macintosh) on the list item to directly enter a custom script name and path. This is useful if the XML file storing the script to event associations is intended for distribution or the script file does not exist on the current machine.

Click on list item	Selects that list item
Shift+click on list item	Adds item to selection
Control+click on list item	Adds/Removes item to/from selection
Double click on list item	Re-associates event to a script file
Option+click on list item (Macintosh) Alt+click on list item (Windows)	Edit in-place the path to the script file

Table 1 InEventScript Association User Actions

Control Strip

The center of the palette (or the bottom if the lower hierarchical list is hidden) contains several controls for manipulation of the palette and modifying script-event associations in the top list.

Figure 2 Control Strip

The left most button is used to show or hide the Hierarchical Event List at the bottom of the palette. Simply

3

click on it to toggle the visibility of the lower portion of the palette.

Also, if the bottom view is visible, the user may click anywhere in the empty area of the control strip and drag up or down to adjust how many rows are shown in the top and bottom lists. Note that no less than three rows of either the top or the bottom of the palette may be shown at any time.

The two buttons on the right edge of the row are used to control the associations in the upper list. The Change Association button (🗂) brings up a file chooser dialog allowing the user to associate a new script with the selected events. If multiple items are selected the same script is associated with each event. The Delete Association button (🗑) removes the association between script and event for all selected items in the upper list. Deleted events are no longer listed in the upper tree. Alternatively, you can hold down the Shift key while clicking the Delete Association button to disassociate the script from the selected event. The event is not removed from the top list and "No Associated Script" displays in the Associated Script field.
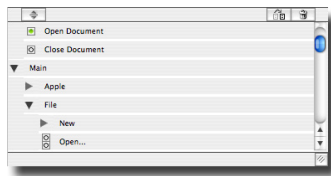

Figure 3 Hierarchical Event View

4

Hierarchical Event View

When shown, the Hierarchical Event View lists all events available for script association. You can associate most events with a script for execution before or after the event occurs. The exceptions are events listed under the Custom heading. Custom events have a single specific timing.

There are two small buttons to the left of each non-custom event name. The upper button represents the before timing and the lower button represents after. Each button has two states. It will either be hollow, showing no script association, or it will be green indicating a script is associated with this event. To associate a script to a certain event, locate it in the tree and click on the dot for the appropriate timing. Table 2 shows what modifier keys can be used to more accurately choose the desired script.

Click	Associate script. A file chooser dialog appears and the chosen file is associated with the event and displayed in the top list.
Shift + Click	Add the event to top list without an associated script. "No Associated Script" is displayed in the Associated Script field.
Option + Click (Macintosh) Alt + Click (Windows)	Add the event to the top list and enter a custom script name. An edit box appears in the Associated Script field. Type in a path to the script file and press enter when finished. A warning icon is shown if the script file does not exist.

Table 2 Hierarchical Event View User Actions

InEventScript Palette Menu

There are four items in the menu, as shown in Figure 4, "Change Script Association", "Remove Script Association", "Run Associated Script" and "About InEventScript…".

The Change and Remove menu items duplicate the functionality of the Change and Remove buttons, respectively.

If you have a single association selected in the top pane, you can run the associated script directly with the Run Associated Script menu item. If errors are encountered, running a script via this menu item will present a dialog with any errors encountered during execution of the script.

The "About InEventScript…" menu item displays a dialog containing information about the InEventScript plug-in. Through this dialog you may also either obtain a permanent license by purchasing the product or obtain a temporary evaluation license. This is also where you enter your registration information to activate InEventScript.
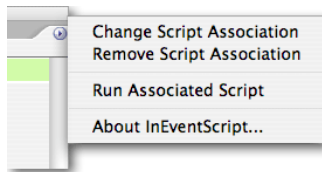

Figure 4 Palette Menu

Custom Events

Launch *(CUSTLaunch)*

The Launch event occurs when the application starts. An associated script will execute at the same time a user could first begin to use controls in the application.

Quit *(CUSTQuit)*

The Quit event occurs when the user chooses to quit InDesign in any way, such as clicking on the quit icon in the application window. The normal InDesign Quit action event only fires when the user chooses the Quit menu item or presses the Quit keyboard shortcut.

Open Document (Before) (*CUSTDocOpen)* / Open Document (After) (*CUSTDocOpen)*

The Open Document (Before) and Open Document (After) events occur any time a document is opened, regardless of how it is opened, such as double clicking on the file in the OS X Finder or Windows Explorer. The normal InDesign Open action event only occurs when the user chooses the Open menu item or presses the Open keyboard shortcut. The Script will execute before the document has begun to open or after the document has completely finished opening, respectively.

Close Document (Before) (*CUSTDocClose)*

The Close Document (Before) event occurs any time after a document is closed, such as during application quit. The normal InDesign Close action event only occurs when the

user chooses the Close menu item or presses the Close keyboard shortcut.

Registration

To register this product you need a valid serial number. To purchase a serial number, go to www.roguesheep.com and look for purchasing information for InEventScript 1.0.3 Alternatively, choose "About InEventScript…" in the palette menu and click "Purchase". This will launch a web browser and take you to our web site where you may purchase a license to enable this product. Another method for obtaining a serial number is to obtain a trial license. Use "Get Trial License" in the About dialog to evaluate InEventScript free of the dialog which will pop up occasionally while in demonstration mode. Once you have obtained a serial number, follow these steps to register your copy of InEventScript:

Figure 5 InEventScript About Dialog

1. Launch Adobe InDesign CS.

8

2. Choose "About InEventScript…" on the InEventScript palette menu.

OR

On Windows: Under the Help:About Plug-Ins:RogueSheep menu, select "InEventScript…".

On Macintosh: Under the InDesign:About Plug-Ins: RogueSheep menu, select "InEventScript…".

3. Click the Register button, and enter the information for "Registered To:" and "Serial:" into the dialog. Make sure you enter both name and serial number exactly as they appear in the e-mail you receive from RogueSheep, including the dashes.

4. Click the Validate button. If you were successful, the dialog reads: "Successfully registered to:" and your name.

Advanced Topic: InEventScriptData.xml

The following section covers the internals of InEventScript and is intended for advanced users and system administrators.

During application startup InEventScript reads the InEventScriptData.xml file. This file contains data associating application events with script files. This data file can be installed on any machine having only the InEventScriptModel plug-in installed. Using this functionality, InEventScript features can be distributed across the various machines at a particular installation without exposing the InEventScript palette to the end-user.

Normally the InEventScriptData.xml file is stored in the InDesign CS application preferences directory. Place the

9

InEventScriptData.xml file with your desired associations in the appropriate directory for each platform to enable InEventScript features without the palette UI.

InEventScriptData.xml Location :

Macintosh

/Users/<user-name>/Library/Preferences/Adobe InDesign/Version X.X/

Windows

C:\Documents and Settings\<user-name>\Application Data\Adobe\InDesign\Version X.X

Previous versions of InEventScript stored the InEventScriptData.xml file in the *InEventScript* directory in the InDesign application *Plug-ins* directory. If you are upgrading from a previous version of InEventScript any data file found at this location will be backed-up to a new file (*InEventScriptData.old)* and then moved to the new location. If you would like to continue using the old location the data file can be edited manually to indicate this preference. See the *ESDocument* element description in the next section of this manual.

Normally the InEventScript palette updates the InEventScriptData.xml file in response to event to script association changes. You can, however, modify the InEventScriptData.xml file outside the context of the InDesign application using a simple text editor.

> *Note: All manual modifications to this file must be done while InDesign is not running.*

InEventScript only reads the elements listed below. All

other information in the InEventScriptData.xml file is discarded. If the file is missing, it will be created after you first make an association in the InEventScript palette.

If the InEventScriptData.xml file is unreadable, an error dialog is displayed at startup and InEventScript will not function properly. Please quit InDesign and make sure the InEventScriptData.xml file conforms to the following XML specification:

XML Elements

ESDocument - This element must exist and contain an ESEvent element for each event to script association.

ESEvent - This element specifies an event/script association and must appear as a child of the ESDocument element to be valid. It has the following possible attributes: type, aid, atype, cname, and path.

XML Attributes

location (data file location) - If you would like to use the old location for the data file (*<InDesign CS Application Folder>/Plug-ins/InEventScript/InEventScriptData.xml*) you can include this attribute with a value of "plugin" on the ESDocument element.

type (event type) - This attribute is required for each ESEvent element. Legal values are: "custom" or "action". If the value is "action", then the attributes aid, atype, and path are also required. If the value is "custom", then the attributes cname and path are also required.

aid (action id) - This attribute contains the value for the desired InDesign action. It is required if the "type" attribute

is "action". The InEventScript download package contains a file named "InDesign CS Actions.txt". This file lists all the available InDesign CS actions and their respective action IDs.

> *Note: Not all actions occur at opportune moments for executing a script, and while we feel a high percentage of these actions may be worthwhile scripting opportunities, some are not. The action may be more specific or more generic than you anticipate. The description of our custom events gives some insight into the side effects you may experience when attaching to the action of your choice.*

atype (action type) - This is the timing desired for the script activation. Legal values are "pre" for before the event, and "post" for the script to run following the user event. This attribute is required if the type attribute value is "action".

cname (custom name) - This is the name of the custom action and is required if the type attribute is set to "custom". These are defined by RogueSheep and include at this time: "CUSTDocClose", "CUSTDocOpen", "CUSTLaunch", "CUSTQuit" and "CUSTDemoLimitReached". For an in-depth look at how these events are used see the section Custom Events, above.

path (script file path) - The value for this attribute is the path, relative to the InDesign application folder, to the script file associated with the event. Double periods may be used to navigate up the folder hierarchy, and front slashes are used as path delimiters. Example: "../../MyScriptFolder/DocOpen.js". This attribute is required for each ESEvent element. When using InDesign CS 1.0, these paths must

12

follow the POSIX convention as is native to Mac OS X and other UNIX platforms, regardless of the platform the InDesign application is running on. When using InDesign CS 2.0, these paths can be either POSIX or Windows style paths regardless of the platform.

Advanced Topic: Suppressing Actions

It is possible to suppress many actions that occur in response to an event with an associated script. This feature can be used to cancel closing of a document or printing for example. InEventScript adds a new method to the Application object in the scripting model that allows the current action to be canceled. Invoking this method in a script associated with Before timing will prevent the action of the event from occurring. See the *OverSetTextWarning.js* demo script for an example use of this feature.

AppleScript	eventscript suppress action
JavaScript	eventscriptSuppressAction
Visual Basic	eventScriptSuppressAction

Figure 6 Suppress Action Method (Application object)

Support

For support please contact us at support@roguesheep.com.

13