

Инструкция по использованию приложения DialogBuilder

Оглавление

Оглавление.....	1
Установка.....	2
Запуск скрипта	2
Предназначение и возможности	3
Совместимость:.....	3
Реализованные возможности в текущей версии:	3
Планируется в следующих версиях:	4
Известные проблемы и недостатки:.....	4
Работа с программой	5
Общие принципы	5
Клавиатурные сокращения:.....	5
Добавление элементов в диалог	6
Редактирование свойств элементов.....	6
Перемещение и удаление элементов	6
Копирование и вставка элементов	7
Редактирование имён переменных	7
Сохранение и контроль результата	7
Открытие файлов.....	8
Работа с настройками программы.....	8
Особенности работы с настройками:	9
Методика и рекомендации разработки диалога в приложении Dialog Builder	9
Дополнительные сведения	10
Обновления в версии 1.82	10
Обновления в новой версии документа относительно предыдущей:	11

Обновлено 03.08.2014

Установка

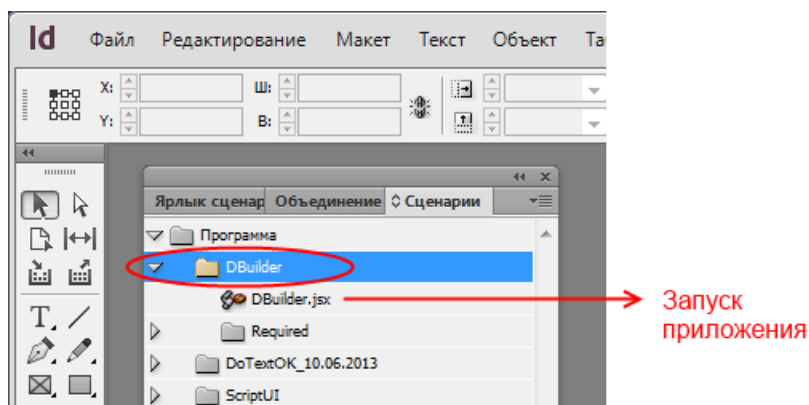
1. Скачайте архив, содержащий последнюю версию программы, из репозитория на сайте **GitHub** — <https://github.com/SlavaBuck/DialogBuilder> (прямая ссылка на архив: <https://github.com/SlavaBuck/DialogBuilder/archive/master.zip>);
2. Из скачанного архива (DialogBuilder-master.zip) распакуйте содержимое папки `contrib` в папку со скриптами Adobe InDesign. (На своей машине с установленной Windows 7 и Adobe InDesign CS6 я использую путь `C:\Program Files (x86)\Adobe\Adobe InDesign CS6\Scripts\Scripts Panel`).

Папка `src` и корневые файлы архива DialogBuilder-master.zip содержат исходный код программы. Для работы с ними требуются библиотеки из репозитория <https://github.com/SlavaBuck/Includes>. Скачайте библиотеки, распакуйте их в отдельную папку со скриптами и в файле `./src/02application.jsx` в строке 12: `#includepath "../Include/"` пропишите к ним корректный относительный путь. По умолчанию предполагается, что все библиотеки находятся в папке `Include`, которая расположена в папке, на уровень выше по отношению к папке с программой (в архиве это папка `contrib`). Если соблюсти данное условие — редактирование путей не потребуется.

Запуск скрипта

Для того, что бы файл скрипта был виден в палитре скриптов Adobe InDesign, его следует располагать по одному из predetermined путей: содержимое папки `C:\Program Files (x86)\Adobe\Adobe InDesign CS6\Scripts\Scripts Panel` отображается в палитре скриптов под группой «Программа» — её можно назвать «системной» папкой скриптов InDesign; содержимое папки `C:\Users\Buck\AppData\Roaming\Adobe\InDesign\Version 8.0\ru_RU\Scripts\Scripts Panel` отображается под группой «Пользователь» — её можно назвать «пользовательской» папкой скриптов. Соответственно, всё то, что расположено в данных папках, будет автоматически видно в соответствующих группах (пути приведены для системы Windows).

Если в «системной» папке скриптов создать папку **Dbuilder** и распаковать в неё содержимое папки `contrib` из архива с программой — в палитре скриптов InDesign у вас должна появиться подобная картина:



Вы можете запустить программу прямо из палитры, выполнив файл `Dbuilder.jsx` из папки `Dbuilder`, или открыть `Dbuilder.jsx` в ExtendScript Toolkit (ESTK) и запустить его из самого редактора.

Предназначение и возможности

Программа предназначена для максимально быстрой и простой визуальной разработки диалогов на языке JavaScript в среде Adobe ExtendScript. На основании разработанной модели диалога формируется «ресурсный» javascript-код, который прямо из программы может быть сохранён в отдельном файле и использован в качестве основы для разработки диалога в целевом рабочем скрипте.

Совместимость:

Программа разрабатывается в среде Adobe ExtendScript CS6 и полностью совместима с версиями Adobe InDesign CS 5.5 и 6 для Windows 7. Предположительно, программа будет нормально работать и с более ранними версиями продуктов Adobe в разных версиях Windows.

В связи со значительными отличиями в системе управления событиями ScriptUI в версии Adobe InDesign CC возможны проблемы (не тестировалось).

Судя по опыту пользователей с операционной системой MacOS — программа не работает для версий Adobe InDesign CS, но частично работает с версией Adobe InDesign CC (при этом инвертируются функции нажатий левой и правой клавиши мыши). Вопрос совместимости с MacOS в текущей версии программы (актуально для версии 1.50) ещё не рассматривался.

Реализованные возможности в текущей версии:

- Генерация корректного javascript-кода для создаваемого макета диалога, с возможностью сохранения и последующего открытия для продолжения редактирования, а также передачи js-кода из приложения в редактор (по умолчанию в ESTK);
- Многодокументный интерфейс — позволяет одновременную работу с несколькими документами-диалогами;
- Полная локализация интерфейса — обеспечивается отдельным файлом locales.jsxinc, расположенном в папке *Requires* и содержащим доступные в пользовательском интерфейсе языки для всех строковых ресурсов приложения (для текущей версии поддерживается русский и английский);
- Поддерживаются широкие возможности по управлению настройками приложения и внешним видом как интерфейса самой программы, так и внешним видом документов-диалогов;
- Процесс редактирования диалога реализован в полном соответствии с технологией Live Update (все изменения настроек редактируемого элемента немедленно отображаются на его внешнем представлении);
- Поддерживается копирование/вырезание/вставка элементов между документами;
- Поддерживается перемещение как отдельных элементов в пределах своих контейнеров (групп и панелей), так и перемещение самих групп в пределах окна диалога;
- Поддерживается возможность управления пользовательскими наборами шрифтов и цветов, которые могут быть сохранены в настройках приложения и использованы повторно;
- Поддерживаются клавиатурные сокращения (к сожалению, корректная работа с клавиатурными сокращениями возможна только при закрытом редакторе ESTK, в противном случае он производит перехват нажатия горячих клавиш);
- Наряду со стандартными ScriptUI-объектами, поддерживается работа с произвольными пользовательскими элементами графического интерфейса, которые могут быть дополнительно включены в файл controls.jsxinc, расположенном в папке *Requires* и, при необходимости, дополнены собственными свойствами (например, так реализована работа с элементом **Separator**).

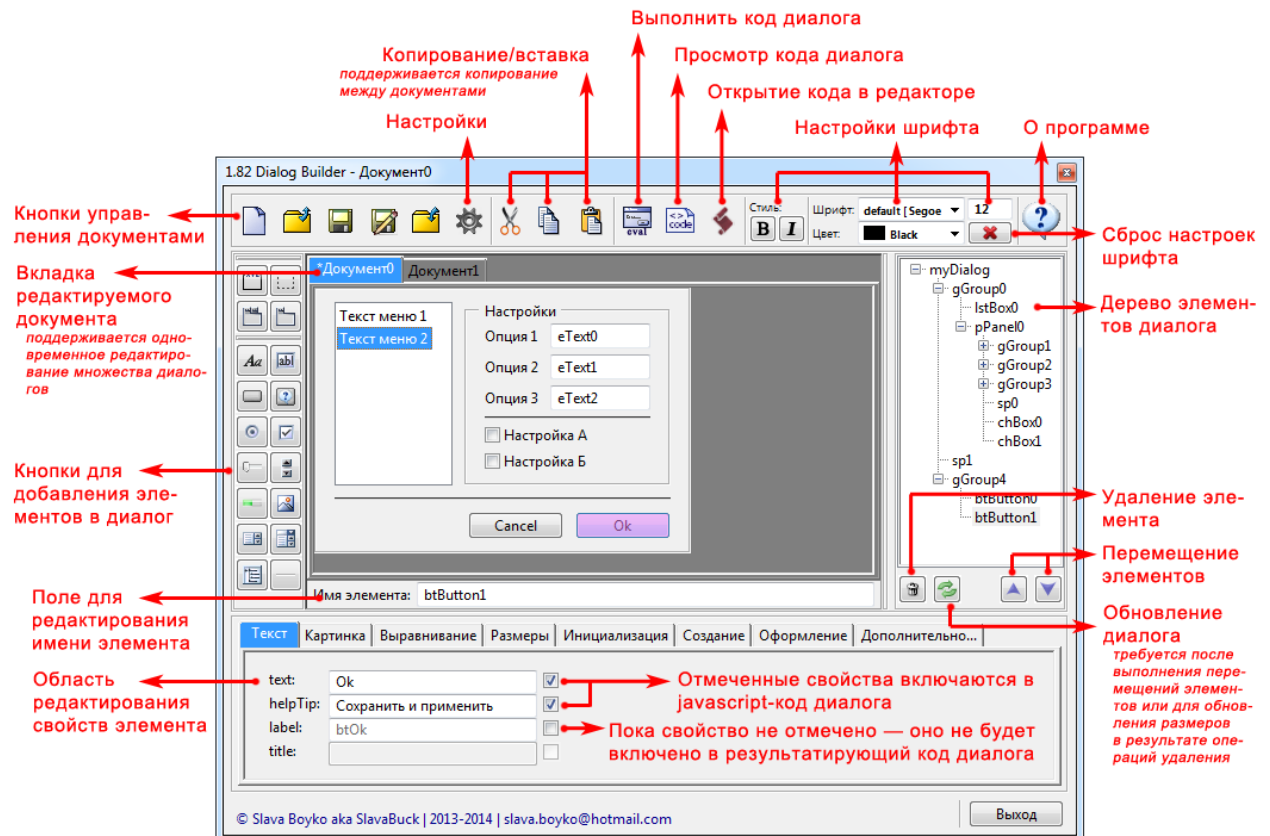
Планируется в следующих версиях:

- Групповое выделение объектов пользовательского интерфейса в документе и выполнение групповых операций: выравнивания, установка размеров и пр...;
- Добавление предустановленных наборов из групп элементов управления для быстрого формирования диалогов на основе типовых шаблонов;
- Расширение возможностей по настройке и работе со списками и изображениями в объекте диалога:
 - конвертация и включение изображений в тело скрипта;
 - настройка масштабирования изображений, помещаемых в пользовательские элементы управления;
 - Работа с массивами изображений для элементов, поддерживающих мультисостояния (IconButton/toolbutton и т.п.);
 - реализация работы с многоколоночными списками и включением изображений в списки;
 - привязка списков к источникам данных;
 - дополнение более широких возможностей по редактированию элемента управления TreeView;
- Использование мастера начального создания диалога с предустановленными шаблонами;
- Внедрение совместимости с MacOS и последними версиями Adobe Indesign CC.

Известные проблемы и недостатки:

- Работа с изображениями и списками реализована на самом базовом уровне, так, например, не реализована работа с многоколоночными списками, настройкой древовидной структуры для TreeView, массивами изображений для элементов с поддержкой мультисостояний и т.п. (подробнее см. в планах не реализацию);
- При генерации ресурсных строк диалогов некоторые цифровые значения свойств сохраняются как строчные значения, например, запись свойства margins:[10, 15, 10, 15] может быть сгенерирована в виде margins:['10', '15', '10', '15']. Эта особенность не влияет на функционирование кода, так как ScriptUI корректно обрабатывает такие случаи и конвертирует эти строчные значения в корректные числовые, поэтому пока эта проблема игнорируется (в том числе из соображений производительности);
- Не совсем корректно вычисляется размер элемента управления TabbedPanel при добавлении в него вкладок (элементов Tab). В связи с нетривиальной логикой расчёта итогового размера TabbedPanel (зависит от многих факторов: кол-ва панелей, их размеров, размеров их подписей и т.д.) исследуются более простые и быстрые возможности это делать, а пока используется грубый и упрощённый подход, который даёт небольшую погрешность в сторону увеличения фактического размера элемента; дефекты размеров в результате редактирования элемента устраняются с помощью операции «обновления/перезагрузки» документа с помощью соответствующей кнопки на панели дерева объектов;
- В отдельных случаях при открытии документов возможны проблемы (в случае, если код диалога содержит инициализирующие строки для элементов-списков, в которых содержаться javascript код), в случае невозможности открытия такого документа нужно очистить код диалога от проблемных строк, открыть и пересоздать (или импортировать их) в уже открытый документ. Далее редактирование можно продолжить и затем сохранить изменения.
- Отсутствие совместимости программы с последней версией Adobe InDesign CC для Windows. Реализация планируется только в долгосрочных планах;

Работа с программой



Общие принципы

Идея написания программы заключалась в предоставлении возможностей максимально простой и быстрой разработки диалогов для скриптов, работающих в среде Adobe ExtendScript и InDesign. Разработка диалога опирается на стандартные возможности компоновщика ScriptUI (AutoLayout manager) при которой размеры и положение элементов диалога вычисляется автоматически в зависимости от их настроек и содержимого. Таким образом, после добавления в документ, элементы требуют минимальной дополнительной настройки (или вообще не требуют). В то же время редактором свойств элементов предусматривается возможность «ручного» управления всеми свойствами элементов, включая их положения в контейнерах и т.п... Кроме того, для удобства редактирования поддерживается работа с горячими клавишами.

Клавиатурные сокращения:

- Ctrl + N** - Новый документ;
- Ctrl + O** - Открыть документ;
- Ctrl + W** - Закрыть документ;
- Ctrl + S** - Сохранить документ;
- Ctrl + K** - Открыть настройки;
- Ctrl + X** - Вырезать;
- Ctrl + C** - Скопировать;
- Ctrl + V** - Вставить;
- Ctrl + R** - Выполнить;
- Ctrl + D** - Показать код диалога;
- Ctrl + H** - О программе...;
- Ctrl + Shift + S** - Сохранить как...;
- Ctrl + Shift + E** - Открыть jsx-файл в редакторе...;

К сожалению, корректная работа с клавиатурными сокращениями возможна только при закрытом редакторе ESTK, в противном случае он производит перехват нажатия горячих клавиш.

Добавление элементов в диалог

Для создания диалога необходимо создать новый документ. С помощью группы кнопок в правой части окна приложения в диалог добавляются все необходимые компоненты. После выделения элемента в нижней части отображаются значения всех доступных для него свойств. Для добавления элементов в контейнер (такие элементы как группы, панели и вкладки / group, panel, tabbedpanel, tab) его необходимо предварительно выделить мышкой (кликнуть либо по самому элементу, либо по соответствующему узлу в дереве компонентов диалога в правой части окна приложения). Чтобы фокус добавления элементов автоматически переустанавливался на контейнер сразу после его добавления в диалог, можно изменить соответствующую установку «Автофокус» на вкладке «Основные» в панели настроек приложения. По умолчанию (без автофокуса) после добавление контейнера, несмотря на то, что он выделяется, активным продолжает оставаться прежний контейнер и добавление новых элементов продолжается в нём.

Редактирование свойств элементов

После добавление элемента в диалог, на него автоматически переустанавливается фокус редактирования, о чём свидетельствует соответствующее цветовое выделение (цвет выделения может настраиваться в настройках: кнопка “Настройки” -> “Основные” -> “Цвет подсветки”) и переключаются поля свойств в нижней части экрана. При этом свойства, не доступные для данного элемента блокируются. Справа от свойства расположено поле для установки флага, который указывает на необходимость включать данное свойство в результирующий код диалога. Свойство не включается в код диалога, пока оно не будет отмечено флажком даже несмотря на наличие в нём какого-либо значения. Для визуального выделения неотмеченных свойств дополнительно используется серый цвет для текста в соответствующих полях редактирования. Редактирование свойств организовано по принципу “Live update” — при котором любое редактирование свойства сразу отображается на внешнем виде элемента. Для выделения любого элемента достаточно кликнуть по нему левой клавишей мыши или выбрать его в дереве элементов, расположенном в правой части окна программы.

Перемещение и удаление элементов

Любой выделенный элемент можно удалить или переместить в рамках своего контейнера. Эти операции также допускаются над целыми группами элементов. Для их выполнения служат кнопки в нижней области дерева элементов (расположено в правой части окна приложения). После начала процесса перемещения элемента все действия с документом блокируются до тех пор, пока не будет использована кнопка обновления документа. После начала процесса перемещения не допускается переключаться между вкладками документа, пока не будет выполнена операция перезагрузки/обновления документа, в то же время перед обновлением документа можно выполнить любое кол-во перемещений для одного или нескольких элементов диалога. О необходимости перезагрузки документа дополнительно свидетельствует красный цвет иконки на кнопке «перезагрузка».

Операция перезагрузки документа также оказывается полезной для нормализации размеров диалога в результате удаления или редактирования отдельных его элементов. В результате выполнения этой процедуры все элементы диалога переинициализируются таким образом, как если бы они создавались в рамках обычного окна ScriptUI. Перезагрузки документа оказывается крайне полезной для нормализации размеров объектов TabbedPanel, а также выравнивания размеров элементов и контейнеров в результате удаления смежных элементов управления (в случаях, когда программа не смогла выполнить это автоматически). Операция перезагрузки документа использует временный промежуточный файл и не отражается на текущем содержимом оригинального файла документа.

Копирование и вставка элементов

Копирование и вставка (в т.ч. и вырезание) осуществляется с сохранением настроек для всех свойств соответствующих элементов. Элементы можно копировать как в рамках одного документа, так и между разными открытыми документами. Копироваться могут как элементы, так и группы элементов с неограниченной глубиной вложенности. Операции копирования/вставки могут выполняться как с помощью горячих клавиш (см. «Работа с программой») так и с помощью соответствующих кнопок в заголовочной панели приложения. При этом, для клавиши «Вставка» предусмотрена всплывающая подсказка, содержащая код элемента, находящегося в данный момент времени в буфере обмена.

Отдельное поведение предусматривается для копирования диалогов целиком: При копировании целого диалога в новый пустой документ — содержимое оригинального диалога помещается в группу (возможно в следующих версиях планируется избавиться от группы и перемещать элементы прямо в диалог). Если на момент копирования диалог-приёмник уже содержит элементы — диалог источник размещается в отдельной группе.

Вставка элементов всегда происходит в «хвост» текущего активного контейнера (группы или панели) в документе-приёмнике. После вставки элементов их, в случае необходимости, можно перемещать в произвольном порядке. После перемещения следует «перезагрузить документ», если перемещение не выполняется — перезагрузка не обязательна.

Редактирование имён переменных

При добавлении элементов в диалог, для них происходит автоматическое генерирование имён переменных. Переменные генерируются на основании шаблона, заданного в настройках на вкладке «Имена переменных». На выбор доступен один из трёх шаблонов: “small” — короткие двух- и трёх-буквенные имена; “full” — длинные имена (установлено по умолчанию); “user” — пользовательская таблица имён (для незадаанных значений используется пользовательское значение по умолчанию, отображаемое серым шрифтом в соответствующем поле элемента). Для генерации имени используется простое правило — к выбранному шаблону имени элемента добавляется его порядковый номер в структуре диалога. Сгенерированное имя переменной доступно для редактирования в соответствующем поле редактора. При редактировании имени переменной для выбранного элемента следует не допускать использование одинакового имени для нескольких элементов в пределах одного контейнера, в то же время элементы из разных контейнеров могут иметь одинаковые имена.

Сохранение и контроль результата

При окончании работы над диалогом его можно сохранить в файл скрипта. В ходе работы можно контролировать его фактический внешний вид с помощью кнопки «Eval» в заголовочной области приложения. Если вид диалога в результате нажатия клавиши «Eval» существенно отличается от редактируемого макета — это, скорее всего, может свидетельствовать о том, что вы забыли поставить галочку напротив одного из изменённых вами свойств у какого-либо элемента (например, изменили но не отметили одно из свойств свойства выравнивания “alignment” / “alignChildren” или размера “preferredSize” / “size” и т.п.). Также, для нормализации размеров элементов можно пользоваться кнопкой «перезагрузки» документа.

Просмотреть текущий код без предварительного сохранения можно с помощью кнопки «Code». Также, можно напрямую передать код диалога в редактор ESTK (при передаче кода происходит автоматическое сохранение и, в случае необходимости, зарос имени файла).

Открытие файлов

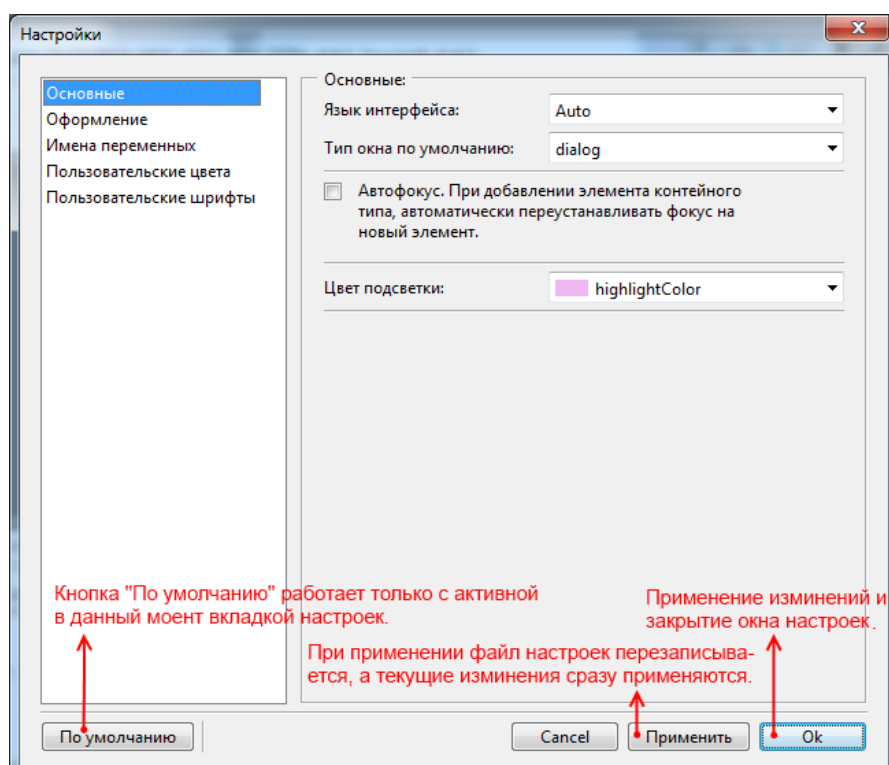
Строго говоря, в программе можно открыть любой javascript файл, содержащий код создания и отображения ScriptU-диалога. Главное, что бы он соответствовал определённым критериям:

- ресурсная строка в объявлении `"var <win> = new Window(...)"` разбита на строки, причём один графический компонент занимает строго одну строку;
- следом за ресурсной строкой следуют блоки с объявлениями локальных ссылок на компоненты диалога и инициализирующим кодом. Блоки между собой разделены пустой строкой. Началом блока выступает объявление вида `"var <control> = <win>.<...>"` за которым могут следовать объявления вида `"var gfx = <control>.graphics; "...` и т.д.
- Последней значащей строкой считается строка `"<win>.show();" , всё остальное игнорируется.`

Работа с настройками программы

Окно настроек доступно по нажатию кнопки «Настройки» в заголовочной области приложения. Первые три вкладки «Основные», «Оформление» и «Имена переменных» содержат все доступные опции для настройки приложения. Вкладки «Пользовательские цвета» и «Пользовательские шрифты» содержат соответствующие таблицы. Все добавленные в них значения становятся доступными в соответствующих элементах управления в рамках всего приложения. Все добавленные в данные таблицы значения сохраняются в файле настроек и остаются доступными после перезапуска приложения.

Нажатие кнопок «Ок» и «Применить» приводит к перезаписи файла настроек и применению внесённых изменений. Нажатие кнопки «По умолчанию» отражается только на открытой в данный момент вкладке.



Особенности работы с настройками:

При запуске программы происходит попытка обнаружения файла *options.jsxinc* в папке Required, расположенной в папке с программой. В случае обнаружения файла (и наличия в нём сигнатуры <DBuilder options>) происходит применение настроек, объявленных в файле. Если файл с настройками не обнаруживается – используются настройки по умолчанию. Файл автоматически создаётся и перезаписывается при применении и сохранении настроек в самом приложении.

Файл допустимо редактировать как с помощью графического интерфейса, предоставляемого приложением, так и в ручном режиме. Все допустимые настройки и их описание содержатся в файле *options.sample.jsxinc*, расположенном в папке Required.

При ручном редактировании файла настроек следует знать, что при сохранении, настройки сохраняются в виде JavaScript-объекта в JSON нотации. Таким образом, в файле они должны синтаксически представлять корректный JavaScript-объект (наиболее частая ошибка – пропуск запятой после определения свойства и другие подобные синтаксические ошибки, специфичные для JavaScript...).

Методика и рекомендации разработки диалога в приложении Dialog Builder

Данный раздел предполагает наличие определённых базовых знаний и некоторого опыта создания диалогов в среде ExtendScript. При разработке и создании макета для диалога следует понимать общие принципы работы системного компоновщика элементов (AutoLayout Manager) для графической подсистемы ScriptUI. Получить базовые навыки работы со ScriptUI можно с помощью замечательно работы **ScriptUI for dummies**, автором которой является Peter Kahrel. Последнюю версию документа можно скачать по адресу — <http://www.kahrel.plus.com/indesign/scriptui.html>, также на сайте <http://adobeindesign.ru/> на странице <http://adobeindesign.ru/2010/12/26/scriptsui/> доступен перевод (более ранней версии) этого руководства (прямая ссылка: <http://adobeindesign.ru/wp-content/uploads/Написание-скриптов.-Пользовательский-интерфейс26.12.2010.zip>).

Приступая к работе над макетом диалога, следует знать, что в пределах контейнера (элемента контейнерного типа, которым выступает один из элементов *group*, *panel*, *tabbedpanel*, *tab* и сам объект диалога: *dialog* или *palette*) элементы располагаются один за другим только по вертикали или горизонтали. Свойство *orientation* в контейнере отвечает за способ размещения в нём элементов. Оно имеет текстовый формат и может обладать одним из трёх значений: *'row'* (элементы располагаются горизонтально в ряд, по умолчанию для контейнеров *group*); *'column'* (элементы располагаются вертикально, по умолчанию для всех, кроме *group*) и *'stack'* (элементы располагаются один над другим).

Для расположения нескольких различных элементов с требуемой ориентацией относительно друг-друга существует два основных способа: 1) Вычислять и задавать требуемые значения свойства *location: [x, y]* для каждого элемента управления; 2) Объединять элементы с помощью контейнеров в группы и размещать такие группы с помощью родительских контейнеров в нужной ориентации. Второй способ является гораздо более простым и наглядным, так как не требует дополнительных вычислений и легко реализуем на практике. Хотя приложение DialogBuilder позволяет использовать оба подхода, именно второй способ является основным в силу своей простоты и наглядности.

Таким образом, общий подход в разработке макета диалога с помощью конструктора диалогов DialogBuilder заключается в создании необходимой структуры вложенных контейнеров с требуемой ориентацией, в случае необходимости контейнеру можно произвольно переустанавливать его свойство `orientation` и, размещением в них требуемых элементов управления.

Проще всего понять вышеизложенный подход позволяет короткое видео на сайте YouTube <http://youtu.be/i6P0OuBvmqI>, в котором продемонстрировано создание небольшого диалога, приведённого в иллюстрации на странице 5 данного руководства.

Вторым, исключительно практическим аспектом, является контроль над размерами определённых блоков диалога или всего диалога в целом, с целью придания ему и его составляющим элементам нужных пропорций, выравнивания и размещения. Тут также возможны варианты. Например, можно задавать нужное свойство `preferredSize: [x, y]` для контейнеров и использовать свойство `alignment` в элементах управления или `alignChildren` – в самих контейнерах. Однако, на практике, практически никогда не нужно использовать заранее предопределённые свойства размеров (`preferredSize` или `size`) для элементов управления контейнерного типа. Вместо этого, следует задавать требуемые размеры самим элементам управления, а расчёт и масштабирование контейнеров оставить на попечении у `AutoLayout Manager`-а. При этом, если элемент управления позволяет задавать собственный размер с помощью свойства `characters` (`StaticText`, `EditText`, ...) — лучше использовать именно его, такой подход избавит вас от неприятностей неправильного размера и неудачного обрезания текста в элементах, при использовании для них различных размеров шрифтов и начертаний.

На практике иногда возникают ситуации, когда в результате удаления элементов или редактирования некоторых их свойств (например это характерно для элемента «`TabbedPanel`») программа не может корректно обновить размеры `ui`-элементов в редактируемом диалоге. В таком случае следует воспользоваться клавишей «Перезагрузка/Reload». В результате выполнения данной процедуры будут переинициализированы все элементы диалога так же, как это происходит в рамках обработки элементов нативными средствами `ScriptUI`-компановщика. Таким образом, элементы примут ту же форму и размеры, какие они будут иметь в рамках создания диалога в результате реального вызова метода `new Window("...")`.

Дополнительные сведения

Текущая версия (1.82) находится в активной разработке, поэтому многие возможности ещё не включены в данную версию.

В остальном программа полностью функциональна и может быть использована для быстрого создания диалогов и импорта их в виде готового JavaScript кода, полностью готового для использования в рабочих скриптах.

Обновления в версии 1.82

- Улучшен алгоритм парсинга открываемых файлов, теперь при открытии файлов, подготовленных или отредактированных вручную (в обход средств приложения по автоматической генерации кода) ошибок стало меньше, при этом сам парсинг стал осуществляется быстрее ;
- Реализована возможность перемещения элементов в диалоге;

- Реализована возможность вырезания/копирования/вставки элементов (поддерживается как в рамках одного документа, так и между разными документами);
- Добавлена возможность оперативной перезагрузки/обновления документа;
- Добавлена поддержка работы с горячими клавишами;
- Обновлена документация и произведён небольшой рефакторинг кода:
 - Устранён вывод в консоль отладочной информации для production-версии программы;
 - Существенно ускорена генерация кода для сложных диалогов (нескольких десятков элементов и более);
 - Программа пересобрана на обновлённой версии библиотек MVC и MVC.DOM;
 - Слегка почищен код и комментарии.

Обновления в новой версии документа относительно предыдущей:

Версия	Описание	Дата	Автор(ы)
1.64	<ul style="list-style-type: none">• В соответствии с новой функциональностью обновлён раздел «Предназначение и возможности»• Добавлен подраздел «Открытие файлов»	15.07.2014	Вячеслав Бойко
1.82	<ul style="list-style-type: none">• В соответствии с новой функциональностью обновлены разделы: «Предназначение и возможности», «Работа с программой», «Контроль и сохранение результатов»• В разделе «Работа с программой» обновлена основная иллюстрация рабочей области приложения и добавлен список горячих клавиш;• Добавлены подразделы: «Перемещение и удаление элементов», «Копирование и вставка элементов», «Обновления в версии X.XX»• Изменён формат таблицы версий файла документации — полная история изменений удалена, теперь в ней содержатся только предыдущие и текущие изменения.	03.08.2014	Вячеслав Бойко

© Вячеслав Бойко, 2014

Г. Киев

slava.boyko@hotmail.com