

Использование «Книги» при вёрстке файлов с концевыми сносками

Я думаю, вы согласитесь со мной, что мы редко выбираем «буду верстать / не буду верстать», если появляется работа на вёрстку, за которую заплатят деньги, то её надо брать и делать. Желательно быстро, и без потери качества.

Это вроде бы очевидные истины, берешь большой файл, делишь его так, чтобы каждая глава в одном файле, и по главам верстаешь. Облачные версии индизайна чем дальше, тем тормознее, но если верстать по главам, то программа работает быстро. А потом собрать отдельные главы инструментом «Книга» (Book) и работа сделана.

Но этот способ не сработает, если в книге есть концевые сноски, инструмент «Книга» о них ничего не знает, и нет способа собрать их в конце книги кроме как загрузить сразу всю книгу, ворочать её всю в процессе вёрстки, и это только для того, чтобы иметь несколько последних страниц с концевыми сносками.

Радости от такой работы не будет много, но дело можно поправить, изменив работу с текстом.

Идея такая:

1) в каждой главе убрать концевые сноски в примечания, оставив на месте каждой концевой сноски её маркер, например, ноль в верхнем регистре.

После этого можно с каждой главой работать как с обычным файлом.

2) когда все главы подготовлены, собрать их все в один файл. У нас будет работа с большим файлом, но на заключительном этапе, когда надо извлечь из примечаний концевые сноски.

3) извлечь из примечаний концевые сноски, удалить из вёрстки примечания и служебные маркеры концевых сносок. Тут останется только работа по стилевому оформлению.

Убираем концевые сноски в примечания

Программа поместит в позиции концевой сноски примечание и служебный маркер, обозначающий, что тут была концевая сноска. Этот маркер — символ в верхнем регистре, по умолчанию это ноль, он примерно равен по ширине удалённому знаку

сноски. Это в определённой мере гарантия, что удаление знака сноски не будет причиной изменения компоновки текста.

В примечание, кроме текста концевой сноски помещается название главы. Третий скрипт из этой последовательности сформирует правильную очередность концевых сносок, такую, что сперва будет название главы, а затем последовательность этих сносок. Но это название главы надо запомнить. Все эти задачи решает скрипт **#1.KeepTextOfEndnoteInNote.jsx**.

В рабочем окне скрипта есть поле ввода названия главы.

Скрипт создаёт группу символьных стилей и в ней символьный стиль, по умолчанию они называются соответственно **'charStyleGroupForEndnote'** и **'endnotePlaceholder'**. Эти названия в общем файле скриптов **'ForEndnoteProc.jsxinc'**.

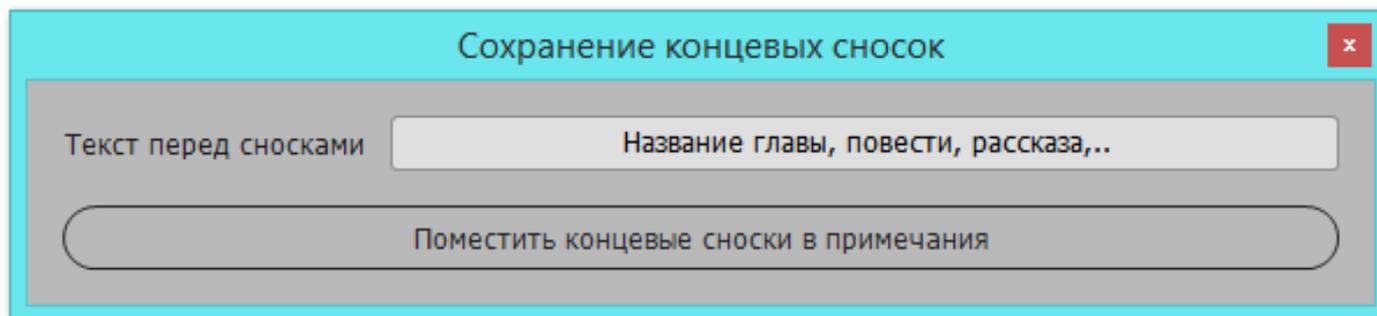
Этим символьным стилем оформляются нули — заместители концевых сносок.

Поскольку в работе могут быть и другие примечания, в тексте примечания между названием главы и текстом концевой сноски стоит разделитель. Этот разделитель в начале текста примечания, тут он в квадратных скобках. Та-

кой, может быть, несколько замороченный способ обезопаситься от размещения текстов из чужих примечаний.

Если концевая сноска в ячейке таблицы, скрипт остановит обработку с сообщением об ошибке. Надо клавишами **Ctrl+Z** откатить вёрстку к состоянию до запуска скрипта, найти концевые сноски в таблице, и вынести их за пределы таблицы. Надо только как-то отметить точки, куда их потом вернуть.

После помещения в примечания всех концевых сносок страница, или страницы, где они были, удаляются.



Сбор файлов Книги в один файл

После того, как вёрстка завершена, и осталось только оформить блок с концевыми сносками,

надо собрать все файлы книги в один файл. Это делает скрипт **#2.MergeBookDocsToOneFile.jsx**.

Название этого файла: 'ItIsAllBook.indd'.

При разработке этого скрипта обнаружилась странная, сложно воспроизводимая ошибка — иногда фреймы в общем файле сдвигаются влево.

Эта проблема решена так: скрипт запоминает координаты текстовых и иллюстрационных фреймов, групп и линий, и после помещения в общий файл у каждого графического объекта его координаты принудительно становятся такими, какими они были в исходном файле. Это гарантирует, что всё останется на своих местах.

Вторая задача — связать текстовые потоки фреймов — простая для решения, т.к. в исходном файле можно получить информацию о связи текстовых фреймов.

Итак, в результате работы этого скрипта из всех файлов книги получается один файл, в котором восстановлены всё текстовые потоки, и все объекты стоят точно на тех же местах, как в исходных файлах.

Теперь осталось восстановить концевые сноски.

Восстановление концевых сносок

Эту заключительную задачу решает скрипт **#3.TextFromNoteToEndnote.jsx**.

В поисках нужных приечаний просматриваются все статьи документа. Из найденных примечаний восстанавливаются концевые сноски и названия глав, к которым они относятся. Обработанные примечания и служебные маркеры сносок удаляются.

Если правка только в сносках

Когда-то я по этой технологии сделал книгу, и думал, что на этом работа завершена. Но через какое-то время мне прислали файл с отредактированными концевыми сносками, там их было за сотню. И попросили «просто заново залить».

Но концевые сноски просто так не «зальёшь». Простая замена одного текста на другой уничтожит связи между номерами и сносками.

Первое, что придёт в голову каждому — теперь надо внимательно выделять новый текст от первого пробела после номера до конца абзаца и брать в буфер. Потом в нужной сноске выделять текст после знака табуляции до конца абзаца и заменять его на содержимое буфера.

Тупая работа, в которой при таком большом числе сносок можно и ошибиться. Но единообразная работа в большинстве случаев легко скриптуется.

И был сделан ещё один скрипт для решения этой задачи. Он называется **#4.UpdateEndnotes.jsx**. Работает он так: в новый файл помещаются новые сноски и запускается скрипт **DoTextOK** для того чтобы отметить символьными стилями курсив, полужирный и пр.

Затем на экране надо открыть два окна с файлами, один с новыми сносками, во втором страница с обновляемыми. Скрипт имеет минималистичный интерфейс, всего две кнопки: '+' и '>'. Ставишь курсор в новую сноску и нажимаешь '+', скрипт сам найдёт текст от первого пробела до конца абзаца, выделит его и возьмёт в буфер.

После этого ставишь курсор в обновляемую сноску и нажимаешь на кнопку '>', программа найдёт текст между знаком табуляции и переводом строки и заменит его на содержимое буфера. Абзацный стиль будет тот, что был у обновляемой сноски.

Тут уже негде ошибиться, и такое перецелкивание сносок похоже на игру.

Итог

Комплект из четырёх скриптов расширяет функционал стандартного инструмента «Книга», теперь в нём можно быстро и удобно верстать книги с концевыми сносками.

Программа объединения файлов Книги в один файл, пожалуй, может быть интересна как отдельный скрипт, поэтому её можно приобрести отдельно.

Михаил Иванюшин
dotextok@gmail.com
dotextok.ru