

Программа DoMakeupOK — приведение вёрстки в порядок. 21.02.2016

Назначение программы

В процессе работы с текстом может появиться множество мелких огрехов, которые верстальщику сложно выловить уставшим от работы взглядом, но если они пролезут в тираж, то будут серьёзными ошибками.

Этот скрипт помогает найти многие ошибки вёрстки, вам уже не нужно будет ломать глаза, прокручивая на экране страницы текста. Это перелистывание по большей части бесполезно, глаз, что называется, «замыливается», и в конце работы мы их уже не видим.

Предполагается, что первоначально текст был обработан скриптом DoTextOK, это гарантирует, что в нём нет вообще уж детских ошибок, таких как повторяющиеся пробелы, дефисы вместо тире, неправильные сокращения «2-ых, 5-ым», и пр. Коль скоро это работа с текстом на финальном этапе, в статьях не должно быть переполнения, и фреймы со статьями не должны размещаться на рабочем столе.

Описание проверок

«Поиск не полностью установленных шрифтов и поиск потерянных глифов»

Эта проверка выполняется так:

- обрабатывается вся статья. Сначала ищутся проблемные шрифты, те, что в меню 'Текст > Найти шрифт...' (в английской версии 'Type > Find Font...' отмечены желтым треугольником с восклицательным знаком. Конечно, верстальщик узнал бы об этой проблеме в момент генерации PDF-файла, но зачем ждать, ведь эта ситуация должна быть решена как можно быстрее.

Если такие «неблагополучные» шрифты обнаружены, скрипт предлагает закончить его работу, решить обнаруженную шрифтовую проблему, а потом перезапустить скрипт.

Перезапуск скрипта после обновления шрифтов обязателен. Если вопрос со шрифтами снят без прекращения работы скрипта, эта ситуация будет обнаружена, и программа сама завершит работу.

- потерянные глифы, если таковые будут найдены, можно вернуть, только если в работе нет проблемных шрифтов.

Глиф — это графический знак символа, и имеющиеся в каждой гарнитуре символы отображаются в панели глифов: Текст > Глифы (Type > Glyphs). Два параметра однозначно описывают такой знак:

1) юникод символа, например, у латинской буквы 'а' он 0061; 2) одно из чисел пары индексов GID/CID: GID (Glyph ID) — позиция или индекс этого глифа в таблице символов конкретного шрифта, CID (Character ID) — идентификатор знака. В этом скрипте используется только индекс GID. Индексы одинаковых по юникоду глифов в разных гарнитурах не совпадают, так для

латинской 'а' в гарнитуре Times New Roman этот индекс равен 68, в гарнитуре Minion Pro он будет 66, в Baskerville значение индекса этой буквы 73. А если глиф в таблице шрифтов отсутствует, то его индекс там равен нулю.

Так, например, если набранном шрифтом Times New Roman тексте есть французские буквы, то при оформлении этого материала гарнитурой Лазурского все французские буквы исчезнут. Но с точки зрения индизайна шрифт с нулевыми глифами не является проблемным. Проблема — вёрстка с такими исчезнувшими символами. Не все потерянные глифы отмечаются перечёркнутым прямоугольником, могут быть случаи, что это просто розовые полоски, исчезающие в режиме предварительного просмотра. И от того, что мы их не видим, эта ситуация становится отложенной ошибкой. Поэтому потерянные глифы (те, что имеют нулевой индекс в конкретной гарнитуре, назовём их нулевыми глифами) надо найти и вернуть в текст. Обычно это делается заменой на сходную по рисунку гарнитуру. В этом скрипте шрифт на замену выбирается не из необозримого списка всех шрифтов приложения, а из шрифтов документа. Поэтому если будут потери и в шрифте с засечками, и в рубленом шрифте, надо для обоих вариантов включить в работу варианты шрифтов, имеющих нужные глифы.

В окне 'Потерянные глифы' отображаются потерянные знаки, и обычно это узнаваемые символы. Но если на месте знака точка или квадратик, то это, скорее всего, символ из шрифта Windings или какого-то другого с глифами-картинками.

Возврат отображения потерянных глифов выполняется в окне «Замена шрифтов нулевых глифов» в выделенном тексте. На время обработки найденные нулевые глифы окрашиваются цветом 'MakeupColor', а при завершении обработки им возвращается цвет, который глиф имел в вёрстке. *Поиск нулевых глифов выполняется во всём документе. По каждому проблемному шрифту выводится информация о числе нулевых глифов в документе и в конкретной статье, но изменения вносятся только в текущую статью. Для поиска других статей с нулевыми глифами надо воспользоваться стандартным поиском глифов.*

«Поиск строк внутри абзаца, начинающихся с дефиса, минуса, тире или знака переноса»

В первой строке абзаца может быть тире, если это диалог, но во всех последующих строках ни дефисов, ни минусов, ни знаков переноса быть не должно. Если не забыть включить вкладку «Шпации в тексте и формулах», то DoTextOK приведёт все дефисы и тире в порядок. Бывают, конечно, частные случаи, когда в присланных на вёрстку текстах вместо буллитов стоит дефис, и эти списки оформлены с использованием принудительного перевода строки. Но в любом случае надо контролировать появление этих символов в начале абзацных строк, и контролировать не просмотром сотен страниц, а при помощи специального инструмента. И такой инструмент есть.

«Поиск абзацев с короткой последней строкой»

В этой проверке ищутся строки с одним словом в последней строке, как с переносом, так и без него. Кроме того, в тексте запускающей программы есть специальный числовой параметр, и он может быть изменен пользователем. Это значение определяет область, в которой может быть и не одно слово, она достаточна для помещения там сокращений «и т.д.» или «и т.п.». Обычно эти сокращения можно без проблем втянуть в абзац.

«Поиск абзацев с почти полными последними строками»

Когда последняя строка абзаца почти полная, маленькое невнятное пустое пространство на правом краю портит вид вёрстки. Вполне можно находить эти случаи и делать выключку на полный формат. Надо только определить допустимую длину свободного пространства последней строки абзаца.

Параметр этой проверки размещён в тексте запускающей программы. Это расстояние от правого края колонки, и если при выравнивании «Выключка влево» последний знак абзаца попадает в эту область, вид выравнивания заменяется на «Выключка по формату». Если этот параметр установлен пользователем в ноль, то эта проверка не выполняется, и её радиокнопка в меню заблокирована.

Для абзацев с выключкой по центру и флагом эта проверка не работает.

«Поиск заголовка внизу полосы»

Эта ситуация нередко бывает в начале вёрстки текста. Но в спешке, или если стилевое оформление невнятное, эти моменты можно пропустить. Переделывать всё равно придётся. Так что лучше подстраховаться от этой неприятности специальным инструментом.

Для оформления книг в формате ePub эта проверка не нужна.

«Поиск висячих строк»

Название говорит само за себя. Ищется первая висячая строка, т.к. информация о том, где дальше ещё есть подобные строки, бесполезна: устранение первой найденной повлияет и на все остальные. Вообще с этой возможностью быстрого поиска вёрстка из рутины становится похожей на игру — охоту на висячек. Для оформления книг в формате ePub эта проверка не нужна.

«Поиск переносов в последней строке страницы/разворота»

Название последней установки поиска изменяется в зависимости от того, где выполняется поиск — на странице или на развороте. При работе с разворотом отмечаются только переносы в последней строке правой страницы, т.к. в случае переноса с левой страницы на правую видны одновременно обе части слова, и дискомфорта при чтении не возникает.

С переносом в последней строке можно при желании расстаться: щелчок на пиктограмме 'цепь' ставит перед последним словом на странице дискреционный перенос. В результате это слово или будет втянуто на текущую страницу, или переместится на следующую. Очень редко, но бывает, что после этого перемещения слова на том же месте появится другое слово с переносом. С ним надо поработать отдельно. Другой возможный случай — после обработки переноса из-за втяжки или разгонки изменяется число строк, что может стать причиной появления висячих строк. Скрипт сообщит об этом изменении числа строк, возникшем на предыдущем шаге обработки. Как вариант решения этой ситуации — откатить на предыдущий шаг, отменив обработку переноса, и попробовать как-то иначе решить эту задачу. Все дискреционные переносы, добавленные этим скриптом, отмечены символьным стилем "DiscretionaryHyphen", и при необходимости их можно все удалить, например, если книга делается ещё и в формате ePub.

Работа программы

Предложенная очерёдность этапов проверки сформирована так, чтобы правка на конкретном шаге по возможности не приводила к необходимости повтора предыдущего шага. К сожалению, это не всегда верно для пунктов «Поиск абзацев с короткой последней строкой», «Поиск заголовка снизу полосы», «Поиск висячей строки», «Поиск переносов в последней строке страницы», поскольку обработка результатов этих проверок может стать причиной изменения числа строк. И это изменение может быть причиной появления висячих строк.

При запуске кнопка «Начать поиск» не активна, поскольку пока не выбрана ни одна из опций. Программа ищет «неблагополучные случаи вёрстки» или во всей статье, или в выделенном тексте. В процессе поиска создается массив номеров страниц, на которых есть по крайней мере одна требующая внимания ситуация. Уточнение «по крайней мере» — это учёт того, что в многоколоннике, например, почти полных строк может быть несколько, но запомнена такая страница будет только один раз.

Требующие внимания строки на время работы скрипта окрашиваются красным цветом. После завершения работы скрипта этот именованный цвет будет удалён, и текст перекрасится в чёрный цвет. Если ищутся висячие строки, абзацы с короткой последней строкой, то предполагается, что пользователь примет решение по избавлению от таких строк; а окрашивание последних строк

абзацев с почти полными строками — это визуальное информирование, какие абзацы были выключены на полный формат.

Переход по страницам выполняется при помощи стрелок вверх и вниз. Перемещение по списку закольцовано: при достижении начала или конца списка в окне отображения информации о текущей странице трижды мигнут строки '<<#>>' и '#####', а затем отобразится конец или начало списка. При работе с висячими строками целесообразно идти по тексту от начала к концу. В других случаях, например, при обработке абзацев с короткой последней строкой лучше идти от конца к началу: если короткое слово будет втянуто, это не повлияет на перетекание предыдущих строк. Любое окно просмотра можно перемещать, и его размещение на экране, в котором была нажата кнопка завершения работы, будет запомнено. При очередном просмотре окно откроется на этом месте.

Михаил Иванюшин