

Назначение программы DoTextOK (23.07.2010)

Причина появления этой программы

Прежде чем начать верстать текст, его нужно подготовить. А освоение компьютера авторами текстов часто становится причиной самых разных ошибок набора. Если раньше машинистка вводила в компьютер текст с машинописной распечатки, и достаточно было убрать повторяющиеся переводы строки, удалить по всему тексту пробел между запятой и тире, и в общем-то всё, то теперь ошибок авторской псевдоверстки стало очень много. Это не значит, что с появлением компьютера стало хуже — нет, теперь не теряются в тексте авторские акценты (курсив, полужирный), верхние и нижние индексы, сноски, и пр. И надо признать, что автор имеет право не знать отличие дефиса от тире, набирать фамилии и инициалы без пробелов между ними, да существует масса ляпов, которые во времена СССР всегда исправлялись при вычитке. К сожалению, сейчас нет того корректорско-редакторского кордона, защищающего издания от ошибок, и иногда авторский сырой текст сразу идет в верстку. Как результат, полно публикаций с дырами между инициалами и фамилией, год вместо «1964 г.» пишется «1964г.», и т.д. и т.п. Можно бесконечно обсуждать эту проблему на форумах, но это никакого результата не даст. Конструктивнее предложить инструмент для всех понимающих проблемы набора, исправляющий ошибки такого плана. Это попытка отделить рутину удаления ошибок набора от творческого процесса верстки и рассказать в ряде случаев, почему текст обработан именно так. Скрипт DoTextOK.jsx надо поместить в каталог Adobe InDesign CS3\Scripts\Scripts Panel и он появится в меню скриптов. Программа работает на платформах PC и Mac.

Конечно, это не единственная программа подобного класса, есть около десятка других, как платных, так и бесплатных, решающих подобные задачи. В этой была сделана попытка оформить в программный код свое понимание подхода к решению проблем набора и объяснить в этой инструкции принимаемые решения. Программа готовит текст к верстке в программе InDesign CS3, InDesign CS4, InDesign CS5.

Варианты обработки текста

1. *Обработка всего текста.* Весь текст обрабатывается, если курсор стоит в текстовом фрейме, но ни один из символов не выделен. Идея программы не только в том, чтобы подготовить текст к верстке. Она еще и информирует о ситуациях, требующих внимания, например, сообщает об обнаружении разделения слов дефисом (минусом) и пробелом — сказать, что должно быть в

данном случае, дефис или тире, можно, только прочитав текст. (Все выводимые сообщения собраны в разделе «Сообщения, выводимые на экран»).

Если встретится абзац, последний символ в котором — запятая, об этом тоже будет выведено сообщение на экран. Может, это стихотворная строка, но если это ошибка набора, то перед версткой ее надо исправить. Критерий, что строка считается оборванной — последний символ абзаца не совпадает ни с одним из этих знаков: . ; : ... — ! ? *. (Проверка на оборванные строки для абзацев, набранных полужирным или курсивом, не выполняется, считаем, что это заголовок).

Иногда автор в тексте иногда курсивом выделяет гласную, на которую падает ударение, эту авторскую разметку тоже желательно не потерять. При подготовке текста в программе Word в большинстве случаев курсив называется *Italic*, а прямой шрифт — *Regular*. Именно эти названия начертаний берутся в расчет при поиске таких одиночных курсивных букв. В процессе работы над этим скриптом DoTextOK.jsx работа над обработкой этих одиночных курсивных ударных гласных выделилась в отдельный скрипт Accent.jsx. После работы этого скрипта одиночные курсивные буквы станут прямыми и будут отмечены стилем myAccent, в котором стандартный черный цвет заменен на myAccentedVocals, по умолчанию он красный — CMYK: 0-100-100-0. При желании после разметки текста в палитре цветов Swatches ему можно назначить другие CMYK-пропорции.

2. *Обработка фрагментов текста.* Если выбрана часть текста, то выборка распространяется до кратности ее абзацам. В этом режиме статистика обработки не выводится.

3. *Подготовка стихов к верстке.* Эти тексты отличаются от обычных тем, что в большинстве случаев повторяющиеся знаки перевода строки не являются ошибкой набора, а служат для членения стихотворения на четверостишия. Для сохранения в тексте этих знаков надо сбросить флажок «Удалить повторяющиеся символы перевода строки».

4. *Доведение до ума чужой верстки.* При решении этой задачи, во-первых, надо привести текст в порядок, во-вторых, быть уверенным, что в результате обработки текста верстка не поползла. Еще не было случая, чтобы в этих работах не использовались одиночные символы перевода строки для дополнительных отбивок в тексте, поэтому надо снять флажок «Удалить повторяющиеся символы перевода строки». Чтобы иметь представление, как повлияет скрипт на верстку, перед запуском DoTextOK имеет смысл окрасить первое и последнее слова (если они есть) в первой и последней строках каждого фрейма при помощи скрипта First&LastWordInFrameInColor.jsx, он в том же архиве, что и DoTextOK.jsx.

О знаке ‘минус’

По каким-то странным стечениям обстоятельств этот знак, имеющий код 0x2212, не попал в кодовую таблицу русских символов. Он отсутствует в гарнитурах BalticaC, BodoniC, FranklinGothicC и пр. Если ввести в текстовый фрейм знак с таким кодом, программа окрасит его розовым цветом, сообщая, что этот символ отсутствует. То, что он есть в стандартных TrueType шрифтах Arial, Times New Roman, Verdana, Georgia, и в двух стандартных PostScript шрифтах Minion Pro и Mugiad Pro, не спасает положение. Но во всех гарнитурах есть короткое тире, ставшее по указанной выше причине распространенным заменителем минуса. Именно этот знак, его код 0x2013, используется в этой программе для обозначения минуса. И даже если в переданном на верстку тексте будет присутствовать «истинный минус» с кодом 0x2212, он будет заменен в процессе подготовки текста к верстке на короткое тире, чтобы при верстке любой гарнитурой в тексте не было розовых букв там, где ожидалось увидеть минус.

Параметры программы

Версия программы — дата последнего обновления, выводимого в её заголовке.

Перед началом обработки текста на экран выводится окно для выбора шестнадцати параметров.

Первый параметр — определяет, как выглядит в тексте тире со стандартными отбивками.

В программе эта величина называется myDashSeparator.

Варианты этого тире:

Длинное тире, обрамленное пробелами, первый из которых неразрывный переменной ширины.

Таковыми будут все тире между словами после обработки текста, тире в начале абзаца будет отделено от следующего слова пробелом, тире в конце абзаца отбито от предшествующего слова неразрывным пробелом. В программе эта величина называется mySpaceDashSpace.

Короткое тире, обрамленное пробелами, первый из которых неразрывный переменной ширины.

В программе эта величина называется mySpaceMinusSpace.

По умолчанию, myDashSeparator = mySpaceDashSpace.

Подготовка текста к верстке (версия 23.07.2010) © Михаил Иванович, 2010

Вариант используемого тире в тексте <input checked="" type="radio"/> обычное тире — <input type="radio"/> короткое тире (минус) —	Разделитель цифр в интервалах годов и чисел <input checked="" type="radio"/> минус <input type="radio"/> тире	<input type="button" value="OK"/> <input type="button" value="Cancel"/>
Вариант сокращений — с точкой или без точки <input checked="" type="radio"/> млн., млрд., трлн. <input type="radio"/> млн, млрд, трлн	Оформление знаков процента и промилле <input checked="" type="radio"/> без отбивки от числа <input type="radio"/> отбиваются от числа тонкой шпацией	
<input checked="" type="checkbox"/> Удалить повторяющиеся символы перевода строки <input checked="" type="checkbox"/> Заменить символы табуляции на пробелы <input checked="" type="checkbox"/> Удалить в тексте символы переносов <input type="checkbox"/> Преобразовать буллиты и нумерацию в текст <input checked="" type="checkbox"/> Удалить гиперссылки из текста <input checked="" type="checkbox"/> Запретить отрывать одно- и двухбуквенные предлоги (частицы, союзы) от слов, к которым они относятся <input type="checkbox"/> Создать символьные стили для курсива, полужирного, верхнего и нижнего индексов <input checked="" type="checkbox"/> Привести в порядок наращение чисел (исправить '1-ых' на '1-х', '2-ого' на '2-го' и т.п.) <input checked="" type="checkbox"/> Присвоить тексту после обработки атрибут "Русский язык"		
<input checked="" type="checkbox"/> Искать в тексте семизначные номера телефонов		
Формат записи телефонного номера <input checked="" type="radio"/> 000-00-00 <input type="radio"/> 000-0000	Разделитель цифр в телефонном номере <input checked="" type="radio"/> дефис <input type="radio"/> пробел	

Второй параметр — определяет, какой знак используется в качестве разделителя между цифрами в интервалах годов и чисел. В программе эта величина называется `myInterDigitSeparator`. По умолчанию задается в качестве этого знака минус с атрибутом `No Break`, можно выбрать тире, оно тоже будет неразрывным.

Третий параметр — вариант сокращений слов миллион, миллиард, триллион. В русском языке есть два правила образования сокращений: 1) путем отбрасывания окончания и завершения слова точкой, тысяч — тыс., 2) выбрасыванием из слова части букв: миллион — млн. Слово «миллионов» сокращается до «млн.» применением обоих этих правил. Примеры такого оформления сокращений есть в любом из томов БСЭ. К сожалению, эти нормы нашего языка вытесняются калькой английских сокращений: `million` — `mln`, и это сокращение у них пишется без точки. Сейчас многих нынешних печатных изданиях встретишь «1 млн» или «2 млрд», именно так, без точки. И коль скоро есть редакторы/корректоры, ратующие как за первый, так и за второй варианты сокращения этих слов, то тут есть возможность выбора.

Другой возможностью использования этого параметра является возможность исправить какую-то неверно набранную ошибку измерения, например, если в тексте сокращение единиц измерения

набрано с точкой «кг.», «мм.» и т.п. Но для этого надо внести правку в код скрипта. Найдите вызов функции `myMlnMlrd`, двумя наклонными чертами прокомментируйте обработку слов, которые надо оставить как есть, и добавьте обработку нужного термина. В тексте для примера оставлено `//myMlnMlrd("кг")`. Обычным поиском/заменой эту замену не сделать: в случае «кг.» это слово может быть в конце предложения, и тогда точку надо оставить. Программа анализирует текст и принимает решение, убрать точку или нет.

Четвертый параметр — выбор, обивать или нет знак процента и промилле от чисел. Тут тоже, к сожалению, не всё однозначно.

Гиленсон (с.423), и Феличи (с.307) считают, что отбивка не нужна. Мильчин в книге 1985 года издания на с. 121 утверждает, что отбивка нужна («Между последней цифрой числа и единицей измерения следует оставлять пробел, в том числе и перед %»).

А что сказано в ГОСТ? На сайте <http://www.pribor.info/docs/?start=0&action=obj&objid=82476&relid=3> есть документ ГОСТ 8.417-2002. ЕДИНИЦЫ ВЕЛИЧИН. Введен в действие с 1 сентября 2003 г. И хотя информация предваряется сообщением «Не является официальным изданием», посмотрим, что говорит этот неофициальный документ. А говорит он в главе 8 Правила написаний обозначений единиц, что эти знаки надо отделять от чисел. Поскольку есть защитники обоих способов оформления знака процента в тексте, предоставим им возможность выбора. Главное, есть возможность оформить всё единообразно.

Пятый параметр — флажок, нужны или нет повторяющиеся символы перевода строки.

Шестой параметр — флажок, оставить в тексте знаки табуляции или заменить их на пробелы.

Седьмой параметр — флажок, удалить или оставить в тексте знаки переносов.

Это символы с кодами 0x00AC и 0x00AD.

Восьмой параметр — флажок, преобразовывать буллиты и нумерацию в текст. Не всегда надо сохранять сделанные в программе MS Word списки. Иногда верстка пойдет проще и быстрее, если они будут представлены как обычный текст.

Девятый параметр — флажок, оставлять или удалять гиперссылки в обрабатываемом файле.

Десятый параметр — флажок «Запретить отрывать одно- и двухбуквенные предлоги (частицы, союзы) от слов, к которым они относятся». Появился по просьбе одного из пользователей, по его словам, на узкой газетной колонке надо снять этот запрет, тогда строки станут плотнее. Может быть... Но в книжной верстке он должен быть установлен, разрыв строки «не пора / ли», «если / бы», одиночные предлоги в конце строки смотрятся ужасно. Стремиться не снимать этот флажок без крайней нужды.

Одиннадцатый параметр — флажок, нужно или нет присваивать символьные стили курсиву, полужирному, верхнему и нижнему индексам. Пользоваться им или нет, решит пользователь. По

умолчанию он сброшен. Целесообразность использования флажка в том, что некоторые шрифты, например, семейства FranklinGothicC имеют только прямое и курсивное начертание. После назначения такого шрифта тексту с полужирным это выделение потеряется. Точно также потеряются верхний и нижний индексы, если назначить тексту абзацный стиль с нажатой клавишей Alt. А этот флажок позволит избежать потери такой авторской разметки текста. При установке флажка курсивный, полужирный текст, нижние и верхние индексы будут иметь свои символьные стили myItalicChars, myBoldChars, myBoldItalicChars, mySybChars, mySupChars, в которых активен только один параметр: черный цвет заменен на myInfoColor.

Двенадцатый параметр — флажок, требуется ли упорядочить наращение чисел. Часто в переданных на вёрстку текстах вместо «3-го» написано «3-его», или вместо «11-го» — «11-ого», и т.п. В большинстве случаев эта процедура полезна, но бывают редчайшие исключения, когда 3-ОГО оказалось каким-то осмысленным сокращением. Чтобы больше не наступать на такие грабли, при установленном флажке скрипт обрабатывает наращения только для строчных букв.

Тринадцатый параметр — флажок, нужно ли присваивать обработанному тексту атрибут «Русский язык». Это присвоение должно выполняться и в русской, и в английской версиях программы. Если назначить атрибут не удастся, то при повторном запуске скрипта этот флажок будет сброшен.

Четырнадцатый параметр — флажок, нужна или нет обработка семизначных телефонных номеров. Если нужна, программа при работе с этими номерами будет руководствоваться следующими двумя параметрами.

Пятнадцатый параметр — формат представления семизначных номеров телефонов 000-00-00 или 000-0000.

В программе это переменная myPhoneSample.

Варианты формата представления 7-значного телефонного номера:

mySample322 = 000-00-00

mySample34 = 000-0000

По умолчанию, выбирается вариант 000-00-00, т.е. myPhoneSample = mySample322

Программа предполагает, что есть только два таких описания семизначных номеров — 000-00-00 и 000-0000. Если появятся другие, нужна доработка функции myRightPhoneNote(). (Программа не любую последовательность цифр трактует как телефонный номер. Если бы это было так, то интервал годов 1917–1924 был бы преобразован в 1917-19-24. При анализе последовательности цифр, похожих на телефонный номер, проверяются предшествующий и последующий символы. Перед и после номера не может быть буквы или цифры, а после номера допустимы только пробел, запятая, точка с запятой и знак перевода строки. При любом другом знаке эта последовательность цифр не считается номером и не

обрабатывается. Идущие подряд семизначные номера будут обработаны, если они разделены знаками пунктуации).

Шестнадцатый параметр — разделитель групп цифр в номере телефона.

В большинстве случаев в качестве разделителя используется дефис. Но в последнее время в рекламных материалах стали встречаться случаи использования немецкого формата разделения телефона, в котором вместо дефиса используется пробел.

mySeparatorIsDefis неразрывный дефис в качестве разделителя

mySeparatorIsDefis неразрывный фиксированный пробел в качестве разделителя

По умолчанию myPhoneDigitSeparator = mySeparatorIsDefis .

Выбранные пользователем установки запоминаются в файле DoTextOK.ini и воспроизводятся при следующем запуске программы.

Цвет myModifiedChars

Программа в процессе работы с текстом иногда удаляет буквы, например, при упорядочении наращения чисел, иногда добавляет пробелы, исправляя такие ошибки набора как 11апреля, IXвек и пр. Очень редко, но всё же бывают случаи, когда надо оставить такое, неверное с точки зрения программы, написание. Поэтому принято решение отмечать все случаи принятия программой решений специальным цветом с именем myModifiedChars. Он тоже черного цвета, как и основной текст, и найти случаи его применения можно в режиме поиска/замены.

Если вы пользуетесь этим скриптом, то сделайте своей практикой первым делом после обработки текста найти все случаи применения цвета myModifiedChars и поправить места, где программа «перестаралась». ☺

Выполняемые процедуры

Добавляются пробелы после ‘г.’ (город) и перед скобками ‘[’ , ‘(’ и единицей измерения температуры °С. (Если пробелов окажется два, программа на следующем шаге приведет все в порядок. Эта процедура гарантирует, что всегда будет пробел перед сокращением и названием города, и при замене перед скобками обычных пробелов на неразрывные ни одна из скобок пропущена не будет. В плане быстрого действия это решение лучше, нежели позже искать ^9[и ^\$(и пр.).

Добавляются пробелы после ‘Дж.’ и ‘св.’. (Если их окажется два, программа на следующем шаге приведет все в порядок).

Заменяем три точки или многоточие после восклицательного и вопросительного знаков на две точки.

Заменяем неразрывные пробелы на обычные. (Обычно это авторская псевдоверстка. Дальше будут выполнены правильные замены обычных пробелов на неразрывные. *Когда в пояснении говорится о неразрывном пробеле, предполагается, что это неразрывный эластичный пробел. Неразрывный пробел добавляется после фамилии перед сокращением имени и между цифрами в телефоне, если выбрано разделение групп цифр пробелами.*).

В зависимости от выбора пользователя — оставляем знаки табуляции или заменяем их на пробелы.

В предыдущих версиях скрипта удалялись символы переноса в тексте. (Такие символы есть в кварковских файлах, если использовалась программа UniSpell или подобная ей. Также удалялись переносы программы WinWord.)

Переписка с пользователями показала, что эти знаки иногда используются, и безусловное их удаление может не ускорить дело, а усложнить работу. Поэтому в скрипте DoTextOK появился выбор — удалять их при обработке или нет. Кроме того, сделаны два отдельных скрипта для удаления каждого из знаков переносов — с кодом \u00AC, и с кодом \u00AD. Они так и называются: Remove_00AC.jsx и Remove_00AD.jsx.

При обнаружении неверной записи даты (есть 25июня 1977г., должно быть 25 июня 1977 г.) вставляем пробел и красим цифру и букву цветом myModifiedChars.

Удаляем повторяющиеся обычные пробелы.

Заменяем два подряд идущих дефиса на тире.

Заменяем истинный минус (знак с кодом 0x2212) на короткое тире.

С этого момента всё, что в обрабатываемом тексте будет по старинке называться привычно минусом, на самом деле будет коротким тире.

Удаляем повторяющиеся маркеры абзаца. (Если установлен флажок «Удалить повторяющиеся символы перевода строки».)

Удаляем пробел в начале абзаца и перевод строки в начале текста.

Пробел и дефис, пробел и тире перед цифрой заменяем на пробел и минус.

Классическая типографика горячего набора рекомендует набирать слитно точку (запятую) и тире.

Но компьютерные шрифты не имеют характерных для горячего набора запяточков, набор плотнее, поэтому в шести следующих заменах принято решение обивать точку или запятую от тире тонкой шпацией.

Замена пробела после запятой перед тире на тонкую шпацию myThinSpace.

Замена пробела после точки перед тире на тонкую шпацию myThinSpace.

Замена дефиса после запятой на тонкую шпацию myThinSpace и тире.

Замена минуса после запятой на тонкую шпацию `myThinSpace` и тире.

Замена пробела и дефиса после запятой на тонкую шпацию `myThinSpace` и тире.

Замена пробела и минуса после запятой на тонкую шпацию `myThinSpace` и тире.

Замена трех точек на многоточие.

Замена точки и многоточия на многоточие.

Замена многоточия и точки на многоточие.

Замена дефиса, отбитого пробелами, на тире, обрамленное пробелами, первый из которых неразрывный переменной ширины. (Далее эта замена называется «тире со стандартными отбивками»).

Замена минуса, отбитого пробелами, на тире со стандартными отбивками.

Замена тире, отбитого пробелами, на тире со стандартными отбивками.

Удаление пробела перед точкой.

Удаление пробела перед запятой.

Удаление пробела перед точкой с запятой.

Удаление пробела перед двоеточием.

Удаление пробела перед восклицательным знаком.

Удаление пробела перед вопросительным знаком.

В зависимости от выбора установок при запуске программы знак процента и знак промилле или ставятся непосредственно после числа или отбиваются от него величиной, присвоенной параметру `myProcSeparator`, по умолчанию, это тонкая шпация.

Удаление пробела перед закрывающей круглой скобкой.

Удаление пробела после открывающей круглой скобки.

Удаление пробела перед закрывающей квадратной скобкой.

Удаление пробела после открывающей квадратной скобки.

Удаление пробела перед закрывающей фигурной скобкой.

Удаление пробела после открывающей фигурной скобки.

Дефис на тире в начале предложения.

Минус на тире в начале предложения.

Ищем случай, когда после восклицательного знака (а также многоточия, запятой, точки с запятой, двоеточия и тире) нет пробела и исправляем его. (вопросительный знак и точка в этот ряд не попадают, т.к. первый символ может быть элементом URL-адреса, а второй может стоять в электронном адресе. В обоих случаях добавление после них пробелов недопустимо.)

Удаление пробела перед маркером абзаца.

Удаление пробела после маркера абзаца.

Неразрывный пробел между цифрой и г., т.е., было например 1954г., будет 1954 г. (тоже для веков).

Неразрывный пробел между цифрой и гг., т.е., было например 1950–1960гг., будет 1950–1960 гг. (тоже для веков).

Неразрывный фиксированный пробел между знаком номера или параграфа и цифрой № 2 § 11.

«Знак градуса от предыдущего числа не должен быть отбит.»

(Из споров о верстке на одном из форумов)

Это спорное утверждение: географические — да, но в градусах измеряют еще температуру. Градусы Цельсия обозначаются °С, т.е. это единица измерения, состоящая из двух знаков, поэтому знак градуса не должен идти сразу после цифры, а вот единица измерения температуры, состоящая из двух знаков, должна быть отбита от цифры.

Почему так сложилось? Не знаю. Да, самая старая книга в моей библиотеке «Справочник корректора» К.И. Былинского и Л.И. Служивкова, Гизлегпром, 1950 на с. 264 рекомендует ставить знак градуса сразу после цифры, и отбивать букву обозначения шкалы пробелом. Но на той же странице пример, в котором в случае записи температуры в виде формулы единица измерения после закрывающей скобки уже написана как °С. Так же оформляется единица измерения температуры как заголовок колонки.

В справочнике «Ручной набор», авт. В.А. Вигдорчик, Москва, «Книга», 1985 на с.63 сказано «Знаки процента, градуса, минуты, секунды нельзя отбивать от цифры, к которой они относятся. Символ, обозначающий шкалу термометра, отбивается от знака градуса на 2 п.»

Другой известный автор П.Г. Гиленсон на с. 423 своей книги «Справочник художественного и технического редакторов», М., «Книга», 1988 рекомендует разбивать единицу измерения, оформляя сведения о температуре так: 12° С.

Возможно, в те времена проще было велеть малообразованному рабочему верстать единообразно знак градуса, чем надеяться, что он поймет, где геометрические и географические градусы, а где температурные.

Но это опыт горячего набора позапрошлого и прошлого веков. На мой взгляд, не должно быть различия в вариантах записи температуры — после числа так, в формулах и заголовках таблиц иначе. Все остальные единицы измерения пишутся единообразно, независимо от того, где используются: с числами, с формулами, или в таблицах. И оформление единиц измерения температуры не должно выпадать из этого ряда. Разумно не разбивать составную единицу измерения, а отбивать всю ее от цифры. Именно так показано в книге А.Э. Мильчина «Справочная книга редактора и корректора», М. «Книга», 1985 г. на с.91, 101. На с. 121 сказано «Между последней цифрой числа и обозначением единицы следует

оставлять пробел, в том числе перед °С». В «Памятной книжке редактора», М., «Книга», 1980, на с. 129 такое же обозначение температуры.

Пересматриваются же правила оформления изданий, например, сокращение слова «страницы» в упомянутой книге К.И. Былинского и Л.И. Служивкова дается как «стр.», но в последних изданиях оно уже стало «с.», почему обозначение температуры так долго стоит особняком?

Кстати, и упомянутый выше ГОСТ 8.417-2002 тоже предписывает набирать знак градуса и указатель шкалы вместе и отбивать эту единицу измерения от числа.

Поэтому надо сделать выбор, как оформлять температуру. И выбор таков: в этой программе °С понимается как единая единица измерения, и отбивается от числа неразрывным пробелом. Без вариантов.

Неразрывный пробел между цифрой и знаком градуса по шкале Цельсия.

Поиск букв в.в., стоящих после цифры (обычной или римской) и пробела, и замена в.в. на вв.

Пробел заменяется на неразрывный.

Поиск букв г.г., стоящих после цифры и пробела, и замена г.г. на гг. Пробел заменяется на неразрывный.

Замена пробел-минус-пробел между цифрами на минус.

Замена пробел-дефис между цифрами на минус.

Замена дефис-пробел между цифрами на минус.

Минусу между цифрами и двум окружающим его цифрам назначается атрибут No Break.

Приведение семизначных номеров телефонов к виду, определенному шаблоном myPhoneSample.

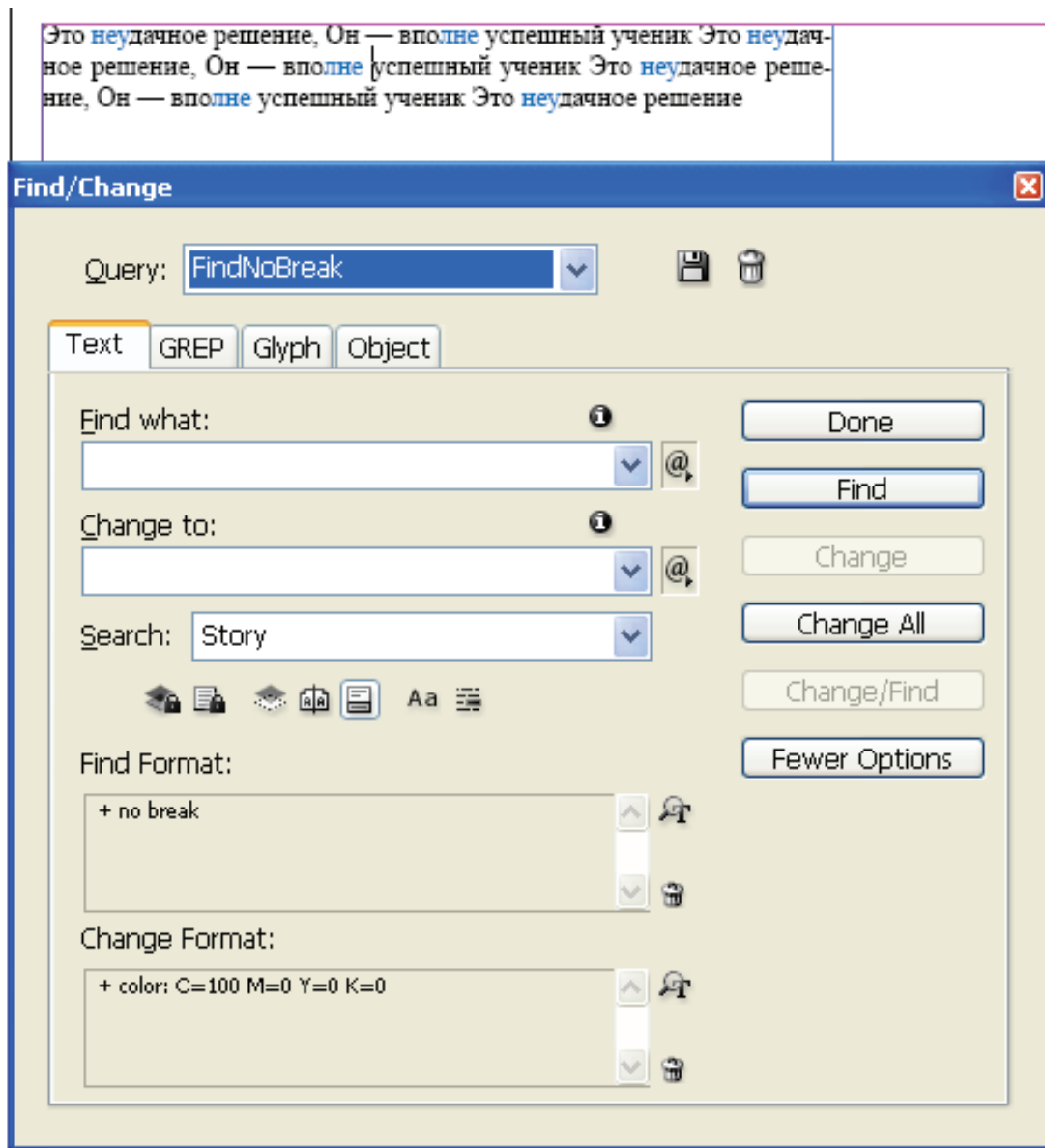
Разделитель цифр — дефис с атрибутом No Break.

Сокращение г. отделяется от цифр года неразрывным пробелом.

Частице 'НЕ' и этим буквам в начале и конце слова присваивается атрибут No Break. Это сделано для того, чтобы, во-первых, не оставлять частицу НЕ в конце строки, во-вторых, исключить возможность переноса как первых букв, так и последних букв НЕ слова на новую строку. Также не рекомендуется оставлять частицу НИ в конце строки, этот случай тоже обрабатывается. «Это не/удачное решение», «Он — впол/не успешный ученик», такие переносы считаются неудачными, и их следует избегать.

Визуализировать символы, отмеченные атрибутом No Break, можно входящим в архив с этой программой запросом FindNoBreak. Этот запрос надо поместить в каталог

Program Files\Adobe\Adobe InDesign CS3\Presets\Find-Change Queries\Text или Program Files\Adobe\Adobe InDesign CS4\Presets\Find-Change Queries\Text\en_US\ . Для его работы должен быть цвет C=100 M=0 Y=0 K=0.



Правила не могут всё предвидеть, и возможны случаи, когда важнее в пределах нескольких абзацев отказаться от правил. Для узкой колонки текста упомянутые выше ограничения для частиц НЕ/НИ могут стать причиной неплотной вёрстки. Поэтому, если не удастся привести абзац в порядок уменьшением трекинга на 5-10 единиц, надо выделить абзац, вывести на экран палитру Character (Window > Type&Tables > Character) и в выпадающем меню снять флажок опции No break. Еще один способ сделать набор плотнее, поступившись в абзаце правилами верстки — если в абзаце есть неразрывные дефисы — заменить их на обычные.

Неразрывный пробел между цифрой и одно-, двух- и трехбуквенной единицей измерения.

Неразрывный пробел между буквой и римской цифрой (чтобы не разрывать Петр I, Людовик XIV).

Замена дефиса между цифрой и буквой на неразбиваемый дефис.

Замена пробел-дефис между цифрой и буквой на неразбиваемый дефис.

Замена дефис-пробел между цифрой и буквой на неразбиваемый дефис.

Во-первых, во-вторых, в-третьих. — обычный дефис (а также 'пробел"дефис' или 'дефис"пробел') заменяем на неразрывный дефис.

Если цифры отделены одна от другой пробелом, он будет заменен на тонкую шпацию.

В сокращениях т.д., т.е., т.ч, т.о, т.к. и т.п., а также когда есть пробел после первой точки — т. д., т. е., т. ч, т. о, т. к. и т. п. — перед второй буквой ставим тонкую шпацию.

Ищутся одно-, двух- трехбуквенные слова и сокращения (а, в, и, к, о, с, у, я, на, не, но, ну, ни, от, из, за, по, до, во, то, см., рис., тт., тов.), пробелы после них заменяются на неразрывные пробелы. (Если любое из этих слов — первое в абзаце, пробелы после них остаются прежними.)

Неразрывный пробел перед словами ли, ль, бы, б, же, уж, ж.

В инициалах между буквами инициалов ставим тонкую шпацию, между фамилией и инициалом — фиксированный неразрывный пробел *.

В случае сокращения имени *Дж.* ставим перед фамилией тонкую шпацию.

Пробел после сокращения «св.» (Святые) делаем неразрывным.

Двум вариантам прямой наклонной черты (SOLIDUS и FRACTION SLASH) одному обратной (REVERSE SOLIDUS) и окружающим их символам присваивается атрибут No Break.

Если в начале абзаца после тире сразу идет текст, он будет отбит пробелом.

Между сокращением г. (город) и названием ставится неразрывный пробел.

В названиях машин, например, Ту-104, обычный дефис заменяем на неразрывный.

В словах, начинающихся с одной буквы, сопровождаемой дефисом — i-операция, N-радикал — обычный дефис заменяется на неразрывный.

Если есть символы, разделенные точкой, после которой нет пробела, то это может быть как часть электронного адреса или адреса сайта, так и ошибка набора. Чтобы при верстке на эти буквы обратили внимание, окрасим их, цвет myInfoColor.

* На мой взгляд, когда инициалы отбиты друг от друга и от фамилии полукруглой, они не смотрятся как единое целое, предложение разваливается. Сейчас стали практиковать набор инициалов без отбивки, в этом есть резон, т.к. точка создает пространство между буквами, но этом случае буквы очень близко друг к другу. Но это вынужденная полумера, обусловленная отсутствием доступа к шпациям разной ширины. В результате проб с разными шпациями мне был выбран вариант, когда буквы инициалов разделены тонкой шпацией, а между инициалом и фамилией стоит неразрывный фиксированный пробел myNonBreakableSpaceFixedWidth. Этот подход к обработке инициалов реализован в этой программе.

Поиск в тексте одиночных курсивных букв,— предполагаем, что это авторское обозначение ударных гласных — выполняется отдельным скриптом Accent.jsx, присваивающий этим буквам символьный стиль myAccent.

Ищем в тексте неправильные наращения, например, 1-ого, 2-ых, и исправляем их: 1-го, 2-х. Все случаи исправления наращений чисел отмечаются цветом myModifiedChars.

Цифровые сноски должны быть отбиты от своих слов, и эта отбивка должна быть заметно меньше пробела. Принято решение использовать для этого тонкую шпацию с атрибутом Superscript.

Ищем известные ляпы набора, например, две запятые (возможно, это попытка обозначить открывающую кавычку), недопустимое сочетание знаков, и отмечаем их стилем myPayAttention.

Всему тексту назначается русский язык.

В процессе отладки вносились исправления, которые, возможно, не нашли отражения в данной инструкции. Поэтому главным критерием, что может программа, должно быть не это описание, а результат работы.

Сообщения, выводимые на экран

Перед запуском этой программы выберите фрейм с текстом или сам текст.

Перед запуском этой программы поставьте курсор в текст для подготовки всей статьи к верстке.

При необходимости привести в порядок уже сверстанную работу выберите фрагмент текста.

Поскольку не исключено, что символы перевода строки использовались для выравнивания текста в полосе, они при обработке фрагмента текста не удаляются.

В фрейме текста нет.

После удаления повторяющихся знаков выяснилось, что текста нет.

Текст к верстке готов.

Фрагмент текста верстки приведен в порядок.

Объем текста до/после обработки [символов:абзацев] —12987:147 / 12971:140. Время работы программы [ч:м:с]: 00:01:17.

1. Неоднозначные варианты разметки текста. Решение надо принять перед версткой:

При обработке текста встречены:

принудительные переводы строки, возможно, это авторская псевдоверстка,

слова, разделенные дефисом и пробелом,
слова, разделенные пробелом и дефисом,
слова, разделенные пробелом и минусом,
слова, разделенные минусом и пробелом,
имеющейся информации недостаточно для решения, что делать с текстом.
Текст будет готов к верстке после исправления указанных ошибок набора.
Повторного запуска этой программы не требуется.

Встречены абзацы, последний символ которых отличается от следующих: . ; : ... — ! ? *.
Такие строки могут быть в стихах или заголовках, в других случаях это, скорее всего, ошибка набора. Для нахождения встреченных оборванных строк перед знаком перевода каретки поставлен пробел.

Встречено неясное использование знаков пунктуации. Эти случаи отмечены стилем myPayAttention (определен только цвет myInfoColor).

Встречены символы, разделенные точкой, после которой нет пробела, это может быть как часть электронного адреса или адреса сайта, так и ошибка набора, например, потеря пробела после точки между предложениями. Они окрашены цветом myInfoColor, и решение, нужен пробел после точки или нет, должно быть принято в процессе верстки.

2. Авторская разметка текста, на которую надо обратить внимание:

(только в случае установки флажка, нужно или нет присваивать символьные стили курсиву, полужирному, верхнему и нижнему индексам)

В процессе обработки текста выделения курсивом, полужирным и курсивом+полужирным, а также верхний и нижний индексы отмечаются своими символьными стилями.

В зависимости от содержимого обрабатываемого файла могут появиться такие стили символов — myItalicChars, myBoldChars, myBoldItalicChars, mySupChars, mySybChars.

Эти стили красят эти слова цветом myInfoColor, чтобы не потерять в процессе верстки при стилевом оформлении текста имеющуюся авторскую разметку.

Предложения и замечания

Если есть идеи или мысли, как доработать программу, чтобы минимизировать ошибки набора, пишите.

Михаил Иванюшин
ivanyushin#yandex.ru
adobeindesign.ru