

1 \Fixing the multi-line link boxes

This package is designed for a Postscript workflow, use `dvips` to convert DVI to PS, then use either Adobe Distiller or Ghostscript to convert to PDF. When building this file, *compile at least three times after changing any hypertext*, there are definitions in the AUX file that need to be brought up to date; otherwise you'll get a Distiller error. There is some package warning that appear in the \TeX log what warn you that some links have not been fully formed. (There are more messages written to the distiller log if you use distiller.)

1.1 Background discussion

The PDF Specification describes and the Adobe Acrobat and Adobe Reader application support multi-line links. The `aebmlink` package was first written in 2006 to bring multi-line links to the \LaTeX world.

1. Let's begin by creating some text to get near right margin: ^{L1}The acrotex website is free to all.
2. Let's begin by creating some text to get near right margin: ^{L2}The acrotex website is free to all.
3. Let's begin by creating some text to get near right margin: ^{L3}The acrotex website is free to all.

Observe the difference between these three links, the first two are in the style originally developed years ago. It consists of a series of little rectangles around each of the syllables of the hypertext string. The rectangles are all linked together so that if you click on one of them they all react. The presence of these little rectangles is usually hidden from the reader by designing a link that is colored, as in the case of the second item above. The link on the *third line* is a new-link; all the little rectangles on the same link are combined together. These are 'true' multi-line links.

1.2 What else is available

The `pdflatex`, `lualatex`, and `xelatex` applications do break a link such as (1) above, but what you get is not a 'true' multi-line link:

1. ^{L4}Let's begin by creating some text to get near right margin: ^{L5}The acrotex website is free to all.

The above link(s) simulate how these applications would operate. They break the text across lines, but the two segments of text are actually two links. Click on one, the other does not react.

For \LaTeX users is the `breakurl` that does break URL-type links (using the `\url` command), but it does not break other types of links.

1.3 The current effort

The current effort is to remove these little rectangles and replace them with one rectangle per line of hypertext. This problem has largely been solved, the results are shown in item 3 above.

To give you an idea of the nature of the problem, the little rectangles of item 1 are internally represented by a series of decimal numbers in groups of eight.

```
ary=[
425 656 442 656 442 664 425 664
447 656 465 656 465 664 447 664
465 656 479 656 479 664 465 664
134 643 150 643 150 652 134 652
151 644 166 644 166 652 151 652
169 644 176 644 176 652 169 652
179 644 195 644 195 652 179 652
198 644 207 644 207 652 198 652
210 644 221 643 221 652 210 652 ]
```

Methodology: Each row represents the bounding box of the syllables of the phrase ‘The acrotex website is free to all’, as determined by \TeX . There are nine rows (syllables), each row has eight entries (quad points). We determine where the text breaks across lines by looking at the second column. The numbers in this column represent the y-coordinates of the lower-left corners of the little rectangles. As you look down that column you’ll notice the numbers change from 656 to 643 as you go from the third row to the fourth row. This tells us that \TeX has broken the line after the third syllable. This observation is fundamental to combining the contiguous boxes that are on the same line.

The challenge was to create Postscript code that manipulates the above array, identify where line breaks occur, pluck out the pairs of numbers what are needed, toss the others away, and load these selected ones into an array that is referenced within the link. Below is the array of number taken from the above array of numbers in a way we need to.

We look at the numbers of the syllables of each line in `ary`, the red elements are loaded into an array in the order they appear, the blue elements are placed at the tail of the entries. The others are discarded. The result of this process is the `aryfixup` array,

```
aryfixup=[
425 656 479 656 479 664 425 664
134 643 221 643 221 652 134 652 ]
```

This is the combined quad points array. Each row is the bounding box of *all the syllables on the line*. The text breaks across lines and exists on two lines, that is why we have two sets of 8 numbers.

2 More Examples

We create links using the various command defined in `aeb.mlinks`.

Creating some text to get near right margin: This is the question that will haunt you for the rest of your life! You ask, “What is this mysterious question?” The hypertext extends over three lines.

Colored links. The acrotex website is free to all. This is the question that will haunt you for the rest of your life! You ask, “What is this mysterious question?”

An URL. We test now `\mlurl` using my uakron web site: <http://www.math.uakron.edu/~dpstory/acrotex.html>

Nameref types. The `\mlnameref` command: \Fixing the multi-line link boxes and its big brother `\mlNameref` ‘Fixing the multi-line link boxes’ on page 1.

3 Acrobat, `web.cfg`: Test wrapping capability

The above title contains font changes, we test the ability to create links to this title, we do this within a `multicols` environment:

Let us generate some text for the purpose of wrapping ‘Acrobat, web.cfg: Test wrapping capability’ on page 3 in a multi-column environment.

Crossing page boundaries, bad In this next paragraph, we demonstrate what happens when a hypertext link crosses page boundaries. The PDF Specification does not support such links, the neither does the user-interface of Acrobat, where you can create multi-line links. `aeb_mlink` detects when a link text crosses a page boundary; it does not create the link, but puts a message in the distiller log (if you use distiller).

We fill the rest of the page with a copy and paste of the same sentence. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. From now on we’ll do normal text. Try this: Let’s write some text that spans a page boundary, we’ll

keep typing until we get the desired page break. We've finally reached the next page. The link is not created. To fix it, you'll have to reword your paragraph to avoid breaking across the page.

3.1 `\bChoices/\eChoices`: Laying out choice fields

From now on we'll do normal text. Move over more, more, more ^{L13}`\bChoices/\eChoices`: Laying out choice fields. Here we demonstrate how to pose a complex section title so it breaks "correctly".

3.2 Other techniques

On some HTML pages I read, I often see multi-line links that are separated by a blank line. We can do that as well.

^{L14}Hello world!

How are you today!

Now we test multi-line links in a `tabular` environment.

^{L15}`\Fixing` the multi-line link ^{L16}`Other` techniques
boxes is a wonderful concept

When `\mlMarksOn` is in force (Section 4), to get the two columns of the `tabular` environment to align as they should, a `\leavevmode` is needed, see the source file.

4 Documentation

The `\mlfixOn` and `\mlfixOff` commands determine whether the little rectangles are combined or not. They affect all `\ml<name>` links within their scope. If expanded within a group, their effects are cancelled when you leave that group. I've also added a new optional key-value within the `eforms` package (dated 2018/03/19 or later) that is only obeyed within a `\ml<name>` link; the key-value pair is `\mlfix{y|n}`. `\mlfix{y}` turns on combining the little rectangles for that link only, while `\mlfix{n}` turns off the combining. The default is `\mlfix{y}`.

4.1 Changing the size of the enclosing boxes

The default height and depth of the boxes are determined by `\strut`, in the current font size. The *new `\mlstrut` key is introduced* (in `eforms` package dated 2018/03/19 or later) for the purpose of changing the vertical dimensions of the rectangles. In the second link below, we say `\mlstrut{\displaystyle\int}`, which gives us the height and depth of a large integral operator. When the link is underlined (`\S{U}`) the depth is set to 1bp, so as to avoid the underlining

appear too low. You can simple increase the vertical dimensions of the boxes by `\mlstrut{\large\strut}` or `\mlstrut{\huge\strut}`; these changes do not affect any text positioning of the content, only the bounding boxes.

Let's begin by creating some text to get near right margin: The acrotex website is free to all $\int f(x)$.

Next line is here

4.2 The user-friendly key-values

The `eforms` package defines user-friendly key-value counterparts to `\mlfix` and `\mlstrut`, these are `mlfix=<yes|no>` and `mlstrut=<strut>`. For example, we test the user-friendly key-value pairs by creating some text to get near right margin.

4.3 The `dbllevel` option and `\mlMarksOn`

The `aeb_mlink` package has new option: `dbllevel=<0|1|2>`, setting `dbllevel` to a positive value yields additional messages in the distiller log (if you use distiller).

When there is a problem with one of the links it may be hard to find the bad link (one that crosses a page boundary). The distiller message gives some hints. The message references a link number, e.g., `mLink58`. By expanding `\mlMarksOn`, links are marked with a superscripted tiny '`L<num>`' (some such marks are seen in [Section 3.2](#) above), so you just search for '`L58`', for example. Turn off this feature with `\mlMarksOff` or simply commenting out the previous `\mlMarksOn`. The default is `\mlMarksOff`. The placement and format of the markers are determined by internal command `\ml@MrkLnk`, which may be redefined as needed.