TEX as a three-stage rocket: Cookie-cutter page breaking

S.K. Venkatesan

Abstract

We trace the progression of technological developments from the early days of computers and how TEX has been able to maneuver through all these changes. A major challenge now arises from the ubiquitous browser and HTML5. We propose that TEX reinvent itself as a three-stage engine: 1) paragraph generation; 2) creation of very long scroll page; 3) page-breaking through a simple cookie-cutter process. With these relatively small adjustments TEX and its typographic nuances could be liberated into the exciting wide open spaces of the World Wide Web.

1 Introduction

Donald Knuth arrived in an age in the United States when universities were making important changes to the digital landscape, notably including Stanford University. The arrival of big capital in software development had already created grave doubts in the mind of the creator of Emacs, Richard Stallman. It is indeed a great achievement that an interactive system with its own special markup, especially for mathematics, introduced in 1979, has been used for 36 years, despite all the advancement of big capital and its great innovations. Knuth's use of the dollar sign to create mathematics of priceless quality is an amusing comment on poor quality commercial typography. Knuth's recent announcement of iTEX [1] was also an apt mockery of the pompous ways of big capital.

However, there have been great changes in the computing landscape. First there were separate non-interactive punch card devices which gave way to interactive command-line interfaces. Second there were automated printing devices and then non-interactive graphics plotting devices. Finally out of all these evolved the modern hybrids: desktops, laptops, tablets, e-readers, mobile phones, etc. It is indeed a great achievement that TEX has survived all these incredible changes in devices and systems. TEX is still probably the best text format to SMS maths in mobiles!

2 SGML, HTML and the browser

Big capital then had its own innovation, the SGML DTD. It could have used a backslash syntax similar to TEX or a parenthetical syntax similar to Lisp; in fact the SGML language was so general that it could accommodate any syntax. Knuth, the mathematician, would not have agreed to use the less-than symbol for markup but that is how it is in the world of business and there is similarly no way a business man would agree to use the dollar sign for markup. Knuth's nice choice of the rarely used backslash for starting macros was replaced with the commonly-used ampersand symbol, which now has to be displayed by escaping it as an entity. However, SGML had an additional feature that TEX didn't have: an agreed-on grammar (EBNF) for parsing, which made it popular with its evangelists. By the late 1980s, SGML+DTD became the industry standard for the large publishers as it could now be validated! However, SGML and the DSSSL stylesheet language for SGML was unwieldy, and it was not easy to process SGML for typesetting. Many commercial companies created SGML-friendly systems with some daring to even implement the unwieldy DSSSL.

The arrival of the Internet produced HTML through Tim Berners-Lee at CERN, a simple implementation of many futuristic concepts of Ted Nelson's Xanadu [2]. With the introduction of the Mosaic browser, the hyperlinking between documents created a new revolution, creating the web as we see it now. It was no more the stale references at the end of a document which one has to look-up in the library; now you can click at links and travel to the linked documents! This led to greater democratisation of knowledge, especially with the arrival of Wikipedia and other open access publishing models, that has continued to expand the open World Wide Web.

Print was still used a lot in those days and it never dawned on anyone at that time that the advent of the browser would slowly bring the end to printing on paper. Internet browsers had a scroll bar that could be scrolled down to read the whole document. Ancient cloth scrolls now had their new avatar. The creation of codex pages that had become the vogue from the 14th century in Europe with the Gutenberg press in Germany is now in the process of being undone.

Of course, the concept of pages is still popular in HTML eBook renderers also. Adobe PostScript and PDF are still popular page models that are used for viewing pages, as the codex refuses to die in the annals of history. The invention of electronic e-paper or e-ink show how friendly glare-free paper is! As the page numbers disappear in eBook Readers like Kindle, they are replaced by the percentage of the document travelled by the reader. Of course, a page number could be generated automatically based on the device width and height, but that would not be a device-independent value any more as in PDF.

3 TEX in a very long browser page

IBM Tech Explorer Hypermedia Browser was one of the early attempts at creating a browser for LATEX markup. It was implemented as plugins for both Netscape and Internet Explorer browsers. Although it only implemented limited features of TEX, it was nevertheless quite useful for viewing scientific documents. Back in 1998, Springer-Verlag's *Linear Functions and Matrix Theory* by author Bill Jacob was one of the first publications that had the privilege of being distributed this way.

Illusions of Java being the platform of the future have faded in recent years, especially after abandonment of the NTS (New Typesetting System). Mainstream development has now shifted to JavaScript with many new implementations of TEX in JavaScript. The JsMath package has been extended by the MathJax team that has now been actively implementing math bits of TEX in the browser. KaTEX is another excellent implementation of TEX math in JavaScript by the Khan academy, which produces popular lectures on all academic topics. The LaTeX-Math.js package tried to implement a complete LATEX document. There is also a complete JavaScript port of pdfTEX (texlive.js) that is now production ready, but it produces PDF on client-side, not web pages!

Boris Veytsman [3] tried to create very long PDF pages using TEX. He used output routines to get the desired long scroll page. However, the maximum limit that TEX can produce is 5 meters! This is also the limit in the PDF specification, a limit that is not easy to relax in PDF, unless we use special units. Thus it is clear that we have to think beyond PDF and into the world of browsers.

The arbitrarily long fluid web pages have the advantage that they can flow into different sized devices, from small hand-held mobiles to wide Desktop screens. The line-breaking algorithms of the browsers have been investigated by the author and compared with the quality of line-breaking that is generally expected out of the TEX system. In the next section we will consider this aspect in detail.

4 Line-breaking: TEX versus browsers

Modern typesetting applications and TEX differ in many details of the line-breaking algorithms. The author has been exposed to a wide variety of commercial WYSIWYG typesetting applications, which make some compromise to render pages faster. However, it is quite possible to produce quality line-breaking from these applications by controlling the default settings of these typesetting systems.

In Figs. 1 and 2 we show some sample paragraphs produced by XELATEX and the Firefox browser to show the differences.

As we can see from this output that by making adjustments in CSS, it is possible to obtain output quite close in quality to that produced by TEX or even surpass it in some cases.

5 Footnotes

Users of eBook reader devices have always been uncomfortable with footnotes becoming endnotes, so that they have to scroll back and forth from its citation, making it inconvenient. The IBM Tech Explorer rendered it as a pop-up notes in those days but there have been other solutions such as sidenotes or a pop-balloon.

The author's own solution for this case[1] has been a bottom rule at the end of the paragraph where the citation of footnote appears.

[1] Footnote below a paragraph

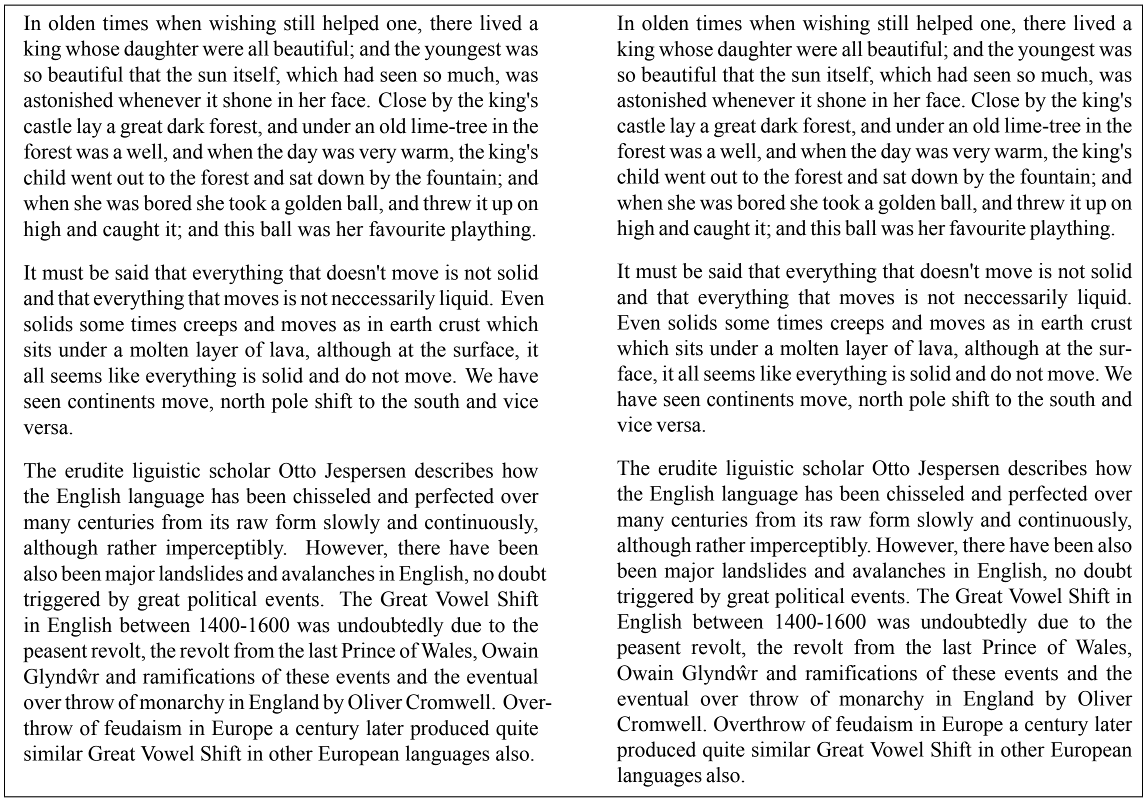
6 Floats: figures and tables

In a bottomless scroll page, where do we place floats, i.e., figures and tables? This again has various "magical" solutions, like pop-ups. The author's simple solution in these cases is again to place it after the paragraph in which it is cited. Of course, it is ideal if the writer of the document indicates the exact location for these floats. Figures and tables can remain at whatever size they are intended, but we can restrict the maximum size to the size of the device, and/or we can add a scroll bar if it is too large, as in the case of wide tables.

7 Pagination as part of a third stage in a TEX browser

It is appropriate at this juncture to create a TEX DVI browser that produces pages as long as there is content without page breaks. The first stage in this process is the module that produces H&J of each paragraph of text. Paragraph creation should be a separate module, as it is much more efficient for an AJAX application to update changes in a paragraph without having to repaint the entire document. Next, we apply the vertical glues, footnotes, floats as mentioned in earlier sections to get the long scroll page.

In a simple streaming-down process it is now possible to automatically decide the page breaks and move the floats one-by-one as we paginate. The floats have already been placed close to their citation in the scroll page, so in a single sweep it is possible to place them as we paginate. Widows and orphans can be controlled by appropriately controlling potentially valid line breaks. We call this simple methodology


In olden times when wishing still helped one, there lived a
king whose daughter were all beautiful; and the youngest was
so beautiful that the sun itself, which had seen so much, was
astonished whenever it shone in her face. Close by the king's
castle lay a great dark forest, and under an old lime-tree in the
forest was a well, and when the day was very warm, the king's
child went out to the forest and sat down by the fountain; and
when she was bored she took a golden ball, and threw it up on
high and caught it; and this ball was her favourite plaything.

It must be said that everything that doesn't move is not solid
and that everything that moves is not neccessarily liquid. Even
solids some times creeps and moves as in earth crust which
sits under a molten layer of lava, although at the surface, it
all seems like everything is solid and do not move. We have
seen continents move, north pole shift to the south and vice
versa.

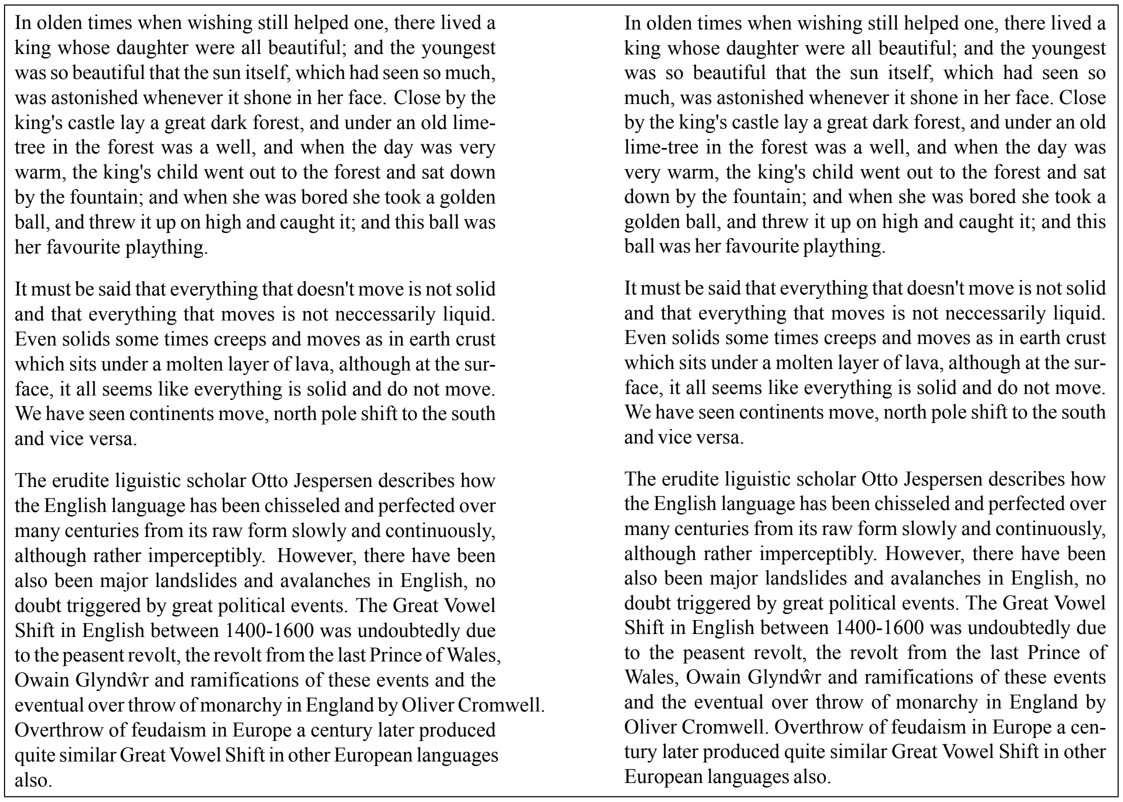
The erudite liguistic scholar Otto Jespersen describes how
the English language has been chisseled and perfected over
many centuries from its raw form slowly and continuously,
although rather imperceptibly. However, there have been
also been major landslides and avalanches in English, no doubt
triggered by great political events. The Great Vowel Shift
in English between 1400-1600 was undoubtedly due to the
peasent revolt, the revolt from the last Prince of Wales, Owain
Glyndŵr and ramifications of these events and the eventual
over throw of monarchy in England by Oliver Cromwell. Over-
throw of feudaism in Europe a century later produced quite
similar Great Vowel Shift in other European languages also.

In olden times when wishing still helped one, there lived a
king whose daughter were all beautiful; and the youngest was
so beautiful that the sun itself, which had seen so much, was
astonished whenever it shone in her face. Close by the king's
castle lay a great dark forest, and under an old lime-tree in the
forest was a well, and when the day was very warm, the king's
child went out to the forest and sat down by the fountain; and
when she was bored she took a golden ball, and threw it up on
high and caught it; and this ball was her favourite plaything.

It must be said that everything that doesn't move is not solid
and that everything that moves is not neccessarily liquid.
Even solids some times creeps and moves as in earth crust
which sits under a molten layer of lava, although at the sur-
face, it all seems like everything is solid and do not move. We
have seen continents move, north pole shift to the south and
vice versa.

The erudite liguistic scholar Otto Jespersen describes how
the English language has been chisseled and perfected over
many centuries from its raw form slowly and continuously,
although rather imperceptibly. However, there have been also
been major landslides and avalanches in English, no doubt
triggered by great political events. The Great Vowel Shift in
English between 1400-1600 was undoubtedly due to the
peasent revolt, the revolt from the last Prince of Wales,
Owain Glyndŵr and ramifications of these events and the
eventual over throw of monarchy in England by Oliver
Cromwell. Overthrow of feudaism in Europe a century later
produced quite similar Great Vowel Shift in other European
languages also.

Figure 1: Rendering at 290pt text width; XeLaTeX on left, Firefox on right


In olden times when wishing still helped one, there lived a
king whose daughter were all beautiful; and the youngest
was so beautiful that the sun itself, which had seen so much,
was astonished whenever it shone in her face. Close by the
king's castle lay a great dark forest, and under an old lime-
tree in the forest was a well, and when the day was very
warm, the king's child went out to the forest and sat down
by the fountain; and when she was bored she took a golden
ball, and threw it up on high and caught it; and this ball was
her favourite plaything.

It must be said that everything that doesn't move is not solid
and that everything that moves is not neccessarily liquid.
Even solids some times creeps and moves as in earth crust
which sits under a molten layer of lava, although at the sur-
face, it all seems like everything is solid and do not move.
We have seen continents move, north pole shift to the south
and vice versa.

The erudite liguistic scholar Otto Jespersen describes how
the English language has been chisseled and perfected over
many centuries from its raw form slowly and continuously,
although rather imperceptibly. However, there have been
also been major landslides and avalanches in English, no
doubt triggered by great political events. The Great Vowel
Shift in English between 1400-1600 was undoubtedly due
to the peasent revolt, the revolt from the last Prince of Wales,
Owain Glyndŵr and ramifications of these events and the
eventual over throw of monarchy in England by Oliver Cromwell.
Overthrow of feudaism in Europe a century later produced
quite similar Great Vowel Shift in other European languages
also.

In olden times when wishing still helped one, there lived a
king whose daughter were all beautiful; and the youngest
was so beautiful that the sun itself, which had seen so
much, was astonished whenever it shone in her face. Close
by the king's castle lay a great dark forest, and under an old
lime-tree in the forest was a well, and when the day was
very warm, the king's child went out to the forest and sat
down by the fountain; and when she was bored she took a
golden ball, and threw it up on high and caught it; and this
ball was her favourite plaything.

It must be said that everything that doesn't move is not solid
and that everything that moves is not neccessarily liquid.
Even solids some times creeps and moves as in earth crust
which sits under a molten layer of lava, although at the sur-
face, it all seems like everything is solid and do not move.
We have seen continents move, north pole shift to the south
and vice versa.

The erudite liguistic scholar Otto Jespersen describes how
the English language has been chisseled and perfected over
many centuries from its raw form slowly and continuously,
although rather imperceptibly. However, there have been
also been major landslides and avalanches in English, no
doubt triggered by great political events. The Great Vowel
Shift in English between 1400-1600 was undoubtedly due
to the peasent revolt, the revolt from the last Prince of
Wales, Owain Glyndŵr and ramifications of these events
and the eventual over throw of monarchy in England by
Oliver Cromwell. Overthrow of feudaism in Europe a cen-
tury later produced quite similar Great Vowel Shift in other
European languages also.

Figure 2: Rendering at 280pt text-width; XeLaTeX on left, Firefox on right.

the cookie-cutter algorithm and have applied for a patent, which is pending approval at the Indian patent office [4]. This algorithm has been tested for hundreds of sample articles and has been found to be quite successful at producing satisfactory paginated results. It is true that the results produced are sub-optimal but are sufficiently close to optimum to be of acceptable quality.

After the fundamental work on automatic pagination by Plass [5], which showed that the optimal solution to the float placement problem is NP-complete, this problem has been reconsidered in a different light by Bruggemann-Klein et al. [6]. Mittelbach [7] implemented a new algorithm in LaTeX 2_ε and Marriott et al. [8] studied the problem for multicolumn layout. However, all these methods are quite complex and involve solving complex optimization problems using a dynamic programming approach. Here, our cookie-cutter method [4] is not trying to solve for a global optimum but instead implements a simple one-pass float placement algorithm that seems to produce reasonable results for a large number of documents.

8 Conclusions

TeX is now facing interesting challenges from the HTML5 browser world. TeX with its own inner beauty and typographic elegance can make the transformation from codex to the scroll page as clearly demonstrated by Boris Veytsman [3]. It now requires the creation of a modern TeX browser, probably with a Unicode model as in XeTeX or LuaTeX, not altogether different from the spirit of IBM Tech Explorer. Doug McKenna [9] has shown that it is possible to rewrite TeX in portable C using his JSBox library. Once we have the box-model output quite like the HTML DOM model, then it is possible to manipulate such box-models with a comprehensive browser in the spirit of HTML5. This will be another exciting browser that will be part of the future world of tablets and devices that can safely carry forward the tradition of TeX into the next few decades.

◇ S.K. Venkatesan
TNQ Books and Journals Pvt. Ltd.
Chennai 600033, India
`skvenkat (at) tnq dot co dot in`

References

[1] Donald Knuth (2010). An Earthshaking Announcement, *TUGboat* 31:2, 121–124. `http://tug.org/TUGboat/tb31-2/tb98knut.pdf`

[2] Theodor Holm Nelson, Project Xanadu. Xanalogical Structure, Needed Now More than Ever: Parallel Documents, Deep Links to Content, Deep Versioning and Deep Re-Use, Keio University. `http://www.xanadu.com.au/ted/XUsurvey/xuDation.html`

[3] Boris Veytsman and Michael Ware (2011). Ebooks and paper sizes: Output routines made easier, *TUGboat* 32:3, 261–265. `http://tug.org/TUGboat/tb32-3/tb102veytsman-ebooks.pdf`

[4] S.K. Venkatesan, M.V. Bhaskar, Srikanth Vittal (2015). Transformation Of Marked-up Content To A Reversible File Format For Automated Browser Based Pagination, Application number 3348/CHE/2015, Indian Patent Office.

[5] Michael Plass (1981). Optimal pagination techniques for automatic typesetting systems. PhD thesis, Stanford University.

[6] A. Bruggemann-Klein, R. Klein, and S. Wohlfeil (1995). Pagination reconsidered, *Electronic Publishing* 8:2–3, 139–152. `http://cajun.cs.nott.ac.uk/compsci/epo/papers/volume8/issue2/2point9.pdf`

[7] Frank Mittelbach (2000). Formatting documents with floats: A new algorithm for LaTeX 2_ε, *TUGboat* 21:3, 278–290. `http://tug.org/TUGboat/tb21-3/tb68mittel.pdf`

[8] Kim Marriott, Peter Moulder and Nathan Hurst (2007). Automatic float placement in multi-column documents, DocEng'07: Proceedings of the 2007 ACM symposium on Document engineering, 125–134.

[9] Doug McKenna (2014). On tracing the trip test with JSBox, *TUGboat* 35:2, 157–167. `http://tug.org/TUGboat/tb35-2/tb110mckenna.pdf`