

The Exotic World of Trade Publishing



Making “Old Media” with the Open Web Platform

Phil Madans & Dave Cramer, Hachette Book Group

Introduction

The Hachette Book Group creates printed as well as digital books using HTML and CSS, the “Open Web Platform.” As a trade publisher, our needs and experiences may be somewhat different than other portions of the industry.

What’s Different About Trade?

Author-Centric

The author’s job is to write. We don’t ask her to structure, format, or design the content, and we have no interest in dictating what tools or methods she uses.

Many Editions of the Same Book

We publish the same book, over and over. A single “work” may be released as:

1. Hardcover
2. ebook (possibly in multiple formats)
3. Large print
4. Trade paperback
5. International edition
6. Mass market paperback

Each of these editions may have a different design, and include different ancillary content.

Obsessed with Typography and Pagination

It would have been far easier to go from typesetting with hot metal directly to HTML+CSS, skipping over the entire era of desktop publishing. Quark and InDesign allows complete control over the appearance of the printed page, and the ability to make nearly an infinite number of changes at little cost. Our business people don’t want to give up that control, and the freedom to micromanage every aspect of the book right until the first pages come off the press.

That need for total control influences everything we do. When we started typesetting with CSS, the single biggest issue we faced was creating an acceptable ellipsis.¹ The look of the printed page is tremendously important, and everyone from the CEO to the production editor has strong opinions about proper typesetting. Hours can be spent talking about the amount

¹ The unicode ellipsis ... is used the the XHTML. A CSS extension prince-text-replace swaps that out for a series of narrow non-breaking spaces and periods.

of space between an open double quote and an open single quote. Some of our divisions insist on thin spaces around em-dashes. Others insist on closed em-dashes.

Every spread must align. Widows are forbidden. Breaks are scrutinized. Every line is judged to determine if it's too tight or too loose.

Profligate with Fonts

The five thousand fonts we have licenses for are not nearly enough! We use swashes, stylistic alternates, and old-style figures. We make extensive use of ornamental and decorative fonts. Even though we publish exclusively in English, we need to set text in Greek, Hebrew, and Arabic.

Simple Books? Complex Designs.

Much of what we publish is fiction or straightforward non-fiction, and appears far simpler than most textbooks. But appearances can be deceiving. A novel may have a dozen different types of extracts—the handwritten note from the serial killer needs to look very different from the text message to the detective. Our mock manuscript (used for testing systems) contains nearly three hundred distinct elements. Footnotes may themselves have footnotes.²

Our Workflow and Tools

Authoring

Authors write in Microsoft Word. Editors and copyeditors work in Microsoft Word. Authors, editors, and copyeditors live by “track

changes,” although even getting to this stage was a struggle for people used to all-paper workflows.

Copyeditors apply structure using Word styles. Only then is the manuscript imported into our production system (IGP:Digital Publisher), and the content converted to structured HTML.

Design and Pagination

Design and pagination happen in IGP:DP, which itself is a web application.

Proofreading and Corrections

PDF proofs of this first pass are created, sent to editorial, and eventually come back with both text and design/flow corrections. Most of the time, these corrections are marked on paper print-outs of the PDFs, and then the same corrections are made by hand (by someone else) to the HTML source. Of course we must proofread again to make sure the corrections were entered correctly.

This is very inefficient. Ideally the proofreader could enter at least the text corrections directly. But we have not found online tools that are as easy and quick to use as an editor's pen on paper. For trade publishing, finding such tools is far more important than better authoring tools.

Final Files

After several rounds of corrections, final PDFs are created and sent to the printer, and the ebook is generated from the HTML content files.

² We publish David Foster Wallace, who is famous for extreme footnoting.

Pain Points

Pagination itself is very labor-intensive

In traditional typesetting, the process of pagination depends on being able to adjust the text layout through various techniques:

1. *Running long or short*—changing the number of lines on the spread to avoid a bad break or widow

We can't yet do this in our systems, as we'd need to dynamically apply a master page with a different margin-bottom.

2. *Tracking*—adjusting letter- and word-spacing to change the number of lines in a paragraph

This is straightforward in CSS, as we use letter-spacing and word-spacing properties.

3. *Adjusting hyphenation parameters*—InDesign and Quark allow you to specify minimum, optimum, and maximum word- and letter-spacing, which gives us control over how strict or lenient the justification will be.

A similar feature has been proposed in CSS 3 Text, but I am unaware of implementations.

4. *Inserting flow characters*—non-breaking spaces, zero-width spaces, and discretionary hyphens can be used to alter the flow of text.

We do this extensively in our HTML/CSS workflow, but must be careful that these characters don't cause problems downstream (in digital books, for example). The discretionary hyphen is particularly problematic, as there does not appear to be consensus on the meaning and displaying of this character.

5. *Adjusting space above and below heads,*

sidebars, images, and tables. This is tricky as any adjustment should apply to the whole spread, not just the page.

Even as CSS typesetting systems support more of these options, it's still a laborious, time-consuming task. TeX and AutoPage³ have solved this problem in the past, why have current tools abandoned this challenge?

Online proofing tools.

Editing is about communication between author, editor, proofreader and designer. In this kind of web-based page design, the compositor, or designer, has become an integral part of the creative team, much more so than in the InDesign page layout workflow. There needs to be simple, intuitive online communication among this group for online content creation to work.

We need a technical solution that doesn't look or feel technical.

Existing design tools have limited utility.

There is also a significant mismatch between the language of book design and the language of CSS. Book designers talk about sinks and baselines, and conversion to margins and padding are nearly impossible, as they depend on details of the font involved.

Font Friction

Identifying the glyph you need in a font, and then getting that glyph to appear in CSS is difficult. We've had to use the InDesign character palette to find GID numbers to insert in CSS via content: prince-glyph-index(1006)

³ AutoPage is a Quark XTension developed by Keith Erf, which automates the layout of complex books to an impressive degree. Interestingly, it's written in LISP.

CSS also is lacking the vocabulary needed to describe modern font families. Many fonts now come with optical sizes, so that a given font may be optimized for varying sizes of text. The Adobe Jenson Pro family contains 32 fonts, but CSS cannot distinguish these:

1. AJensonPro-Capt.otf
2. AJensonPro-Regular.otf
3. AJensonPro-Subh.otf
4. AJensonPro-Disp.otf

A font-optical-weight property would be useful:

```
font-optical-weight: subhead;
```

Font families also have uncategorizable variants. Antique Olive contains a “Nord” variant. Bermuda has four weights: solid, dots, squiggle, and open. The Knockout family (illustrated at right) has 27 weights, which merge the concepts of font-weight and font-stretch.

Running heads and generated content

Running heads that contain any change in type properties (italic, different font) are awkward to create.

One book needed a running head that consisted of a page number, an ornament, and the book title or author name. This had to be centered on the page, and so needed to be in `@top-center`. But the ornament used a different font. So I needed to use flows in conjunction with pseudo-selectors, and embed some useless markup in the HTML to make it all work:

```
@page body:right {
  @top-center {
    content: flow(recto);
  }
}
```

Knockout No. 26, Junior Flyweight. This narrowest, most compact member of the Knockout family, was t
Knockout No. 26 Junior Flyweight

Knockout No. 27, Junior Bantamweight. The light and narrow styles of this range
Knockout No. 27 Junior Bantamweight

Knockout No. 28, Junior Featherweight. Sharing the matter-of-fact
Knockout No. 28 Junior Featherweight

Knockout No. 29, Junior Lightweight. Ideal for informat
Knockout No. 29 Junior Lightweight

Knockout No. 30, Junior Welterweight. Compos
Knockout No. 30 Junior Welterweight

Knockout No. 31, Junior Middleweight. Se
Knockout No. 31 Junior Middleweight

Knockout No. 32, Jr Cruiserweight. R
Knockout No. 32 Junior Cruiserweight

Knockout No. 33, Jr Heavyweight
Knockout No. 33 Junior Heavyweight

Knockout No. 34, Junior Sumo
Knockout No. 34 Junior Sumo

Knockout No. 46, Flyweight. These ultra-compressed members of the Knockout family thrive at very large
Knockout No. 46 Flyweight

Knockout No. 47, Bantamweight. Sufficient space appears in this font's counters t
Knockout No. 47 Bantamweight

Knockout No. 48, Featherweight. Subheads, sidebars and pullquote
Knockout No. 48 Featherweight

Knockout No. 49, Lightweight. Bold enough to attract the
Knockout No. 49 Lightweight

Knockout No. 50, Welterweight. Without overpov
Knockout No. 50 Welterweight

Knockout No. 51, Middleweight. The open
Knockout No. 51 Middleweight

Knockout No. 52, Cruiserweight. Kno
Knockout No. 52 Cruiserweight

Knockout No. 53, Heavyweight. F
Knockout No. 53 Heavyweight

Knockout No. 54, Sumo. Page
Knockout No. 54 Sumo

Knockout No. 66, Full Flyweight. Banner headlines need supercompressed forms for their extra large sizes.
Knockout No. 66 Full Flyweight

Knockout No. 67, Full Bantamweight. Despite its width, this font works in remarkab
Knockout No. 67 Full Bantamweight

Knockout No. 68, Full Featherweight. An adaptation of American wor
Knockout No. 68 Full Featherweight

Knockout No. 69, Full Lightweight. For headlines and cove
Knockout No. 69 Full Lightweight

Knockout No. 70, Full Welterweight. Packing the p
Knockout No. 70 Full Welterweight

Knockout No. 71, Full Middleweight. Suitec
Knockout No. 71 Full Middleweight

Knockout No. 72, Full Cruiserweight
Knockout No. 72 Full Cruiserweight

Knockout No. 73, Full Heavywei
Knockout No. 73 Full Heavyweight

Knockout No. 74, Full Sumo.
Knockout No. 74 Full Sumo

Knockout No. 90, Ultimate Welterweight. Not mei
Knockout No. 90 Ultimate Welterweight

Knockout No. 91, Ultimate Middleweight. E
Knockout No. 91 Ultimate Middleweight

Knockout No. 92, Ult. Cruiserweight.
Knockout No. 92 Ultimate Cruiserweight

Knockout No. 93, Ult. Heavyweig
Knockout No. 93 Ultimate Heavyweight

Knockout No. 94, Ultimate Su
Knockout No. 94 Ultimate Sumo

```
p.recto-cus {  
  flow: static(recto);  
  content: prince-glyph-index(80);  
  font-family: 'Type Embellishments One';  
}
```

```
p.recto-cus:before {  
  content: 'Book Title';  
  display: inline;  
  font-family: 'Garamond 3 LT Std';  
}
```

```
p.recto-cus:after {  
  content: counter(page);  
  display: inline;  
  font-family: 'Garamond 3 LT Std';  
}
```

Named pages

In WYSIWYG typesetting, master pages are used extensively. CSS provides some basic support for this concept, using named pages and the `:left`, `:right`, and `:first` pseudoselectors. We would like to dynamically assign master pages. For example, books with run-in chapters should not have a running head if the chapter title falls at the top of a page. The ninth and tenth pages

of a chapter may need a different bottom margin to fix a paging problem.

What Can W3C Do?

1. Finalize versions of relevant CSS3 specifications—text, fonts, paged media.
2. Support implementations of those relevant CSS3 specs.
3. CSS needs more options for running heads.
4. CSS could use additional facet(s) for describing font variants.
5. W3C should sponsor a feature inventory, where publishers and CSS working group members can identify gaps in current and proposed specifications.
6. Ultimately, we want to produce print and digital content with similar workflows, tools, and standards. The IDPF and W3C should work together to avoid divergence between standards—we don't want to do EPUB running heads with XSL-FO and print running heads with CSS paged media.