

Teaching Publishing as Information Technology: A Description of the Structural Editing Curriculum at a South African University

Liam Borgstrom

Biographical paragraph:

Liam Borgstrom is a lecturer in Publishing Studies in the Department of Information Science at the University of Pretoria, in South Africa, where he specialises in books production and workflow.

Abstract:

As traditional publishing becomes increasingly digitalised the responsibilities of its professionals become more shared out. This is especially true in situations where copy-editors are required to work within a mark-up (increasingly XML) paradigm. We have found in the development of our curriculum that we have shaped our teaching of editorial practice to consistently reinforce the underlying structure of a publication, as inferred from the technology and practises of a production department. As such, the philosophy of creating a publication from an informed structural perspective has become one of our key driving concepts reinforced over the course of several modules in our degree and CPD programmes.

Introduction

At the University of Pretoria we have hosted a Bachelor of Information Science with a Specialisation in Publishing programme since 2000.¹ Over the years the programme has benefitted from a shifting staff with various focuses of expertise and for the past ten years

has included a module focussing specifically on digital publishing. Within our programme we have, almost, inadvertently (through the construction of the programme) developed a system in which structural editing, mark-up, and standardisation are being taught throughout our modules no matter the intended output. As a result, the hope is to impress on our graduates the mind-set of modern publishing practice, without limiting them to making use of a single system or approach. It is the author's opinion that publishing can be presented in many ways, as book and literature history, as business and communications, and—as the author now sees it to be important—as information technology.

Modules within the Programme

At the University of Pretoria the publishing studies undergraduate programme is a three-year degree (conferring a BA: Publishing) begun in 1996.¹ It is multidisciplinary, incorporating marketing, visual arts, information science, languages, and publishing-specific modules.² At the first-year level, students are introduced to the industry as a whole and its place within the South African economy. At second-year level the modules focus on practical aspects of copy-editing and editorial project management as well as typesetting and reproduction. At third-year level, students are introduced to commissioning practices for books and magazines, to electronic publishing, and to book marketing. Also offered is an honours degree and a research masters and doctorate.

This analysis is focussed on the theory taught at the undergraduate level, where the standardised practical theory contains within it the basis for the skill set required for working in a digital environment with the architectural mind-set and technological tools of a structural editor.

Concept of Structural Editing

Structure and architecture are inherent in books. This was true from the early days of printing, but these considerations can now be built into the design of content at the stage of editing. 'The book is an extension of the eye,'³ and the relationship between page and reader is symbiotic: the reader gives purpose, while the page gives meaning.

In an age of human-machine interaction we need that meaning, in its visuospatial presentation, to be predefined in order to allow for interactivity and customisation. Editing structurally requires that we define the function of every item of content within an intended publication. In doing so, from an editorial standpoint, this allows for the editor to function as an architect, and it provides a logical framework for the editor to work within.

To illustrate this, one could say that an architect has two primary responsibilities: One, to the client, to ensure an attractive, safe and usable space, suitable for their needs. Two, to the construction crew, to ensure that each worker understands what materials should be used, where the plumbing should be laid, and how it should be decorated, and so forth.

An editor operating structurally must do the same, and consider the order that the information is presented in, the language in which it is conveyed, as well as placing headings, and assigning supplementary content. To the production department, the editor should be able to present this information in a way which is suitable to their operations, by using specific and controlled vocabulary that efficiently and accurately describes the function and the intended formatting of a given piece of content.

Good editing should be invisible, and allow the reader to effectively navigate their way through the publication.

The Teaching of Structure

Beginning in second year with PUB 210 we begin teaching the role of copy-editing. This course is taught from a practical and managerial point of view. The idea is to give the editor a top-down view of the publication, not just as a piece of prose, but rather as a collection of multiple content types.

Our introduction to copy-editing looks at the copy editor not so much as a language editor but as a manager whose main purpose is managing the editing and content creation phase, while also liaising between the various contributors to publication. Beginning with editorial theory,^{4,5} we divide the processes of editing into micro- and macro-editorial functions. Micro-editing is our content-specific function that involves language editing, detailed editing for sense, and marking up the manuscript. Macro-editing is done to process the overall structure of a publication and is referred to as content editing, substantive editing, and briefing.

Practically this is implemented in the assessment where our students are required to submit a project containing an editing manuscript as well as a series of briefs.⁶ The briefs represent the 'macro' logic, and are marked first, as they demonstrate the students' arrangement of content, i.e. how they have indicated text to be captioned, glossary words, sidebar content, and so forth. These briefs outlay the plan and intended final form of the publication, as it is applied to the draft manuscript, they are the blueprints. The manuscript itself is then used to assess the students' application of micro-editing in applying the specifications (structural, grammatical, and logical) of briefs.

The model for this project mostly derives from local practices, where this module—developed by local professionals—has always included manuscript mark-up as a means to ease the burden of the typesetter.

I must digress here to explain how we have differentiated this method in terms of what is usually considered to be mark-up.

Pure language editing and re-writing is taught within modules in our English department. When we speak of mark-up in the context of South African publishing, we are not describing the use of symbols by which we can indicate typographic forms—such as small caps or regularised spacing—though these are part of it; rather we borrow our use of the term from information technology. As our course was developed in an era of computer-based typesetting, and because our module has become compulsory for multimedia students within our Department of Information Science, we employ a manner of mark-up that is more like XML.

That is to say, we teach our students to make use of two separate style-sheets: one for the purposes of standardising language, spelling, numbers, punctuation, and so forth; and another that is intended for the typesetter, which describes the means by which styling should be applied, for example, as illustrated in Figure 1:

H1: 18pt
Times, **bold**

H2: 14pt,
Times, ***bold italic***

Int: Introductory paragraph
Preferably indented and distinctive from genl body text

Defl: Definition list
To be used in glossaries, may be set as body type
Should indicate term in **bold**
Definition in roman

A/W#: Will reference number as indicated in attached brief
Please see brief for sizing and caption information
Unless otherwise indicated, artworks should be rectangular

Foot: Please set this information in the footer accordingly

Figure 1. An example of a typographic style-sheet accompanying a manuscript. This serves as a legend to explain the tags inserted in the manuscript, as well providing basic formatting instructions.

From a historical perspective this is necessary, as the symbols many editors have been taught were developed for use in the context of proof correction. It has been common practice in South Africa for copy editors to insert an intermediate phase in the development of the first proofs. Given the amount of desktop editing that is practised, and the digital nature of typesetting systems, during the developmental years of this module many typesetters were given plain text files to import into design programs such as Quark Express. As a result, it was imperative to provide a clean digital typescript and a printed typescript with the structure of the publication visually encoded in it according to the typographic style-sheet.

Today, our students have the ability to work completely digitally—although we do insist on physical pen and ink editing—as well as the possibility of publication via a modern XML system. Hence we have in our mark-up a similarity to such systems where a manuscript may be provided with, e.g., <DEFL> tags. The purpose for this on a physical level is primarily instructive to the designer; however, on a subliminal level the practice encourages our students to think as a Web developer would. In computer-readable code, all entities should be defined. From this perspective it is possible to make use of word-processor styles for import into typesetting systems, as well as planting the seeds in our students for future practices in digital publishing.

With this module we aim to have students seeing the book from a macro-level in which they are able to envision the book holistically before it goes into production. Much of our publishing industry is dependent on educational schoolbook publishing, where, in order to speed up production times (often a book is required to be completed within four to six months), much of the book material may be sent to the author predesigned, with delineated space for text. Pressure from the industry therefore necessitates that new

publishers think structurally. In physical books the need to stick to page lengths and to comply with government requirements for content necessitate that the book is thought out in its entirety at the point of commission.

For digital books, making use of a structural and semantic tagging system is essential for XML- and HTML5-based publication outputs. Publishers intending to make use of new EPUB standards would do well to insist that their editors understand the concept of HTML semantic mark-up.

In 2015 we introduced for the first time a short course for the training copy editors in HTML development for EPUB. The course, set out over two days, instructs copy editors on how to edit within an HTML document according to the demands of a CSS (Cascading Style Sheets) style-sheet. In this way, a Web developer is able to construct a broad plan for publication according to a brief and to technical specifications, such as understanding that supplementary sidebar content should be indicated by an `<aside>` tag. This tag may have numerous classes, e.g., `<aside class="DYK">`, and even relate to EPUB specific mark-up, e.g., `<aside epub:type="sidebar" class="DYK">`. For this reason we believe it is fundamental to train students to be capable of thinking not just of an author's manuscript but of the publication as an organized grouping of content types.

In teaching this particular course we have the delegates (our descriptive term for the student's role in this course) build their own textbook. The language is unedited, and the content describing actions is without mark-up. Delegates are taken through the first 'chapter' step by step, and, as they follow the manuscript, they apply mark-up accordingly. Beginning with simple categorisation of text, they then gradually apply the tags to content areas, include attributes, and also insert imagery and multimedia content.

On the second day they are given an extensive lesson on table formatting, using appropriate tags and classes in conjunction with the predesigned CSS, not only to set material in a table, but to define, accurately, header rows, total rows, and so forth. In the environment of an adaptive educational textbook, JavaScript can be used to hide or reveal defined content based on a learner's actions. Therefore, in an exercise on categorisation, items such as table header cells can be revealed for self-assessment. What is more, the accurate mark-up of content makes the application of continued customisation and editing simpler through the use of 'search and replace' tools as well as regular expressions.⁷

The benefit of this type of workflow carries over into the production of print and digital books through layout software such as Adobe InDesign. In this case, efficient use of this software requires a structural mind-set. Throughout my stewardship of the our second-year module, PUB 220 (Publishing Design and (Re)Production), it has been made part of the assessment in teaching layout software that the layout file itself—the proprietary INDD (a file format exclusive to Adobe InDesign software) as opposed to the publishable PDF—be examined and the use of in-built styling features be used to their full effect. This helps to create a facilitative working environment where each content type is accurately defined, allowing for more efficient editing later on, as well as for consistent editing during the formative phase. Modern features of the software now allow for HTML tag mapping specifically for digital output.

When considering a visual (print) layout for digital reproduction, the structure of the document is essential. The nature of mark-up languages is hierarchical, and therefore they require that the proper nesting occurs within each and every tag. It is possible to do this in a WYSIWYG (what you see is what you get) format as supplied by the software. However, it is

not merely a matter of applying it; it is essential for the student to understand that, from the nesting of mark-up, the visual intention should be accurately exported as, for example, in Figure 2:

```
<aside class="DYK">
  <figure>
    
    <figcaption id="ch1aw3c">The classification system
    that we use today was developed in 1735 by Carl
    Linnaeus.</figcaption>
  </figure>
</aside>
```

Figure 2. An example of hierarchial mark-up in HTML. This will be interpreted by a browser or reading application.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi posuere non ipsum quis ornare. Nulla facilisi. Donec porttitor blandit risus eget iaculis. Donec arcu tellus, feugiat vel congue in, vehicula sit amet dolor. Aliquam ac ante tellus. Interdum et malesuada fames ac ante ipsum primis in faucibus. Morbi blandit lectus augue, eu venenatis nisl blandit ut. Donec euismod eu ipsum vitae pulvinar. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Suspendisse potenti. Sed sit amet nunc sed purus tincidunt ultricies. Nunc venenatis hendrerit tellus dictum rhoncus. Curabitur sem magna, hendrerit vel lacinia ut, accumsan eu ipsum. Ut viverra, odio vitae congue imperdiet, sem sem dapibus est, nec rutrum eros quam in leo. Aliquam bibendum luctus ante quis tincidunt.

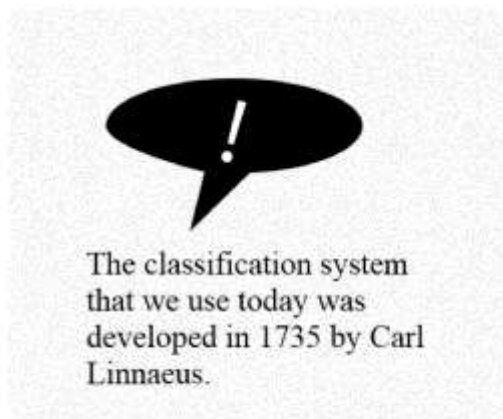


Figure 3. An example of the above code as displayed in a browser.

Conclusion

With all the possibilities that information technology brings to modern publishing, structural editing seems to be a valuable if not an inevitable way of working. This at least has been made apparent to us in the publishing studies programme at the University of Pretoria in our local context of educational publishing—with its tight deadlines and high-pressure, high-demand work environments, combined with a government mandate to produce better digital learning solutions. Publishing successfully in such a context requires that we train editors who are capable of working in a digital environment, and this requires more than language skills; our graduates must be able to cope and function as structural information architects.

Notes

1. Elizabeth Le Roux, 'How Should We Teach Book Publishing in This Time of Digital Transformation? A View from the South' (paper presented at By the Book: The Book and the Study of Its Digital Transformation, Publishing Studies Conference, Florence, Italy, 2014)
2. University of Pretoria BIS Publishing programme, <http://www.up.ac.za/information-science/article/1821944/bis-publishing>
3. Marshall McLuhan and Quentin Fiore, *The Medium is the Massage: An Inventory of Effects* (Berkeley, CA: Gingko 1967)
4. Barbara Horn, *Copy-editing* (London: Horn Editorial Books 2008)
5. Judith Butcher, Caroline Drake, and Maureen Leach, *Butcher's Copy-editing: The Cambridge Handbook for Editors, Copy-editors and Proofreaders* (Cambridge: Cambridge University Press 2006)

6. Individual briefs are drawn up for an illustrator, and designer, as well as a list of comments and queries intended for the author and commissioning editor, and finally a letter to the head of production explaining the state of the publication's development.

7. Regular expressions refer to a controlled vocabulary used to automate search and replace actions. Many text-editors support regular expressions and could use it afford broad standardised changes in mark-up. The following example is a regular expression used to find e-mail address: `\b[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,}\b`⁸

8. Jan Goyvaerts, Regular-Expression.info, <http://www.regular-expressions.info/email.html>