# Automated Analysis Example: Moby-Dick

## Introduction

AH Formatter V7.1 is able to automatically detect a range of typographic problems in a formatted document. Solving these problems usually requires editorial or stylistic changes, and sometimes both. Automated analysis of formatting problems is most useful with longer documents. With shorter documents, the user might decide they can find all of the problems just by looking at the few pages.

The example that is discussed here is a pastiche of the first American edition of Moby-Dick by Herman Melville. Moby-Dick was chosen because:

- Moby-Dick is frequently used as a sample document for EPUB and CSS examples.
- At around 650 pages when formatted, it is obvious that automated analysis will be both quicker and more consistent than visually inspecting each page.
- The book is out of copyright.
- The text is freely available in XML.
- Scans of the original pages are available on the web. (1) (2)

This document describes analysis and correction occurring in separate stages for different error types. This is simply for ease of explanation. In practice, errors can be corrected in any useful sequence.

## Automated Analysis

Languages, including English, have stylistic conventions for formatted text. The origins of the conventions may be for readability, for aesthetics, for commercial reasons, or for a mix of these. Some are now just considered to be good design without reference to the underlying reason. Books on typography or book design will usually cover a subset of possible problems, but even the reference books differ in what they consider to be a problem, the threshold for a condition becoming a problem, and even the terminology for describing a problem.

Automated analysis (3), introduced in AH Formatter V7.0 and expanded in V7.1, can detect a range of error conditions:

- Too many blank pages at the end of the document

  The printing and binding method used for a book may require that the book is a multiple of 8, 16, 32, or even more pages. Extensions to the force-page-count property make this possible with AH Formatter V7.1. However, the forced page count can result in empty pages at the end of the document just to fulfil the requirement. Empty pages are a cost to the publisher with little or no obvious benefit.

- Too many consecutive lines end with a hyphen

  Too many consecutive lines that end with a hyphen increase the likelihood that a reader will either skip reading a line or read the same line twice. Both the Chicago Manual of Style (17th edition) (4) and Elements of Typographic Style (5) recommend a maximum of three consecutive lines that end with a hyphen.

- Too many consecutive lines that all start or all end with the same word

  This is similar to the problem with multiple consecutive lines that end on a hyphen. Multiple consecutive lines that start with the same word or multiple lines that end with the same word can result in a reader either skipping a line of text or rereading a line. The Chicago Manual of Style (17th edition) recommends a maximum of three lines that either start or end with the same word. Book Typography (6) warns against multiple lines that end with the same word but does not provide a limit and does not mention lines that start with the same word.

- Lines before or after current block

  When a chapter does not start on a new page, there can be a requirement for a minimum number of lines either before or after the chapter heading. Book Typography recommends at least three lines above and below the chapter heading. This can usually be enforced using the widows and orphans properties, but not when, for example, the previous chapter ends with short lines of dialogue.

- Page widow

  A short last line of a block of text that is formatted as the first line on a page or column can affect readability.

- Paragraph widow

  A short last line of a block of text can affect readability. A secondary consideration is that many paragraph widows can add extra pages, and cost, to a document.

- River

  A river occurs where spaces on consecutive lines overlap, or nearly overlap. Rivers are more likely to occur in justified text than in text that is aligned to one side or is

centered. A large or long river of white-space may interfere with comprehension of the text. People differ in their sensitivity to rivers, but it is often noted as problem for people with certain cognitive disabilities, including dyslexsia.

- Unbalanced spread

  It can be an aesthetic requirement that text blocks on facing pages are the same length. However, the First Edition has multiple unbalanced spreads.

- White-space

  Excessive white-space between words can affect readability.

## Analysis report

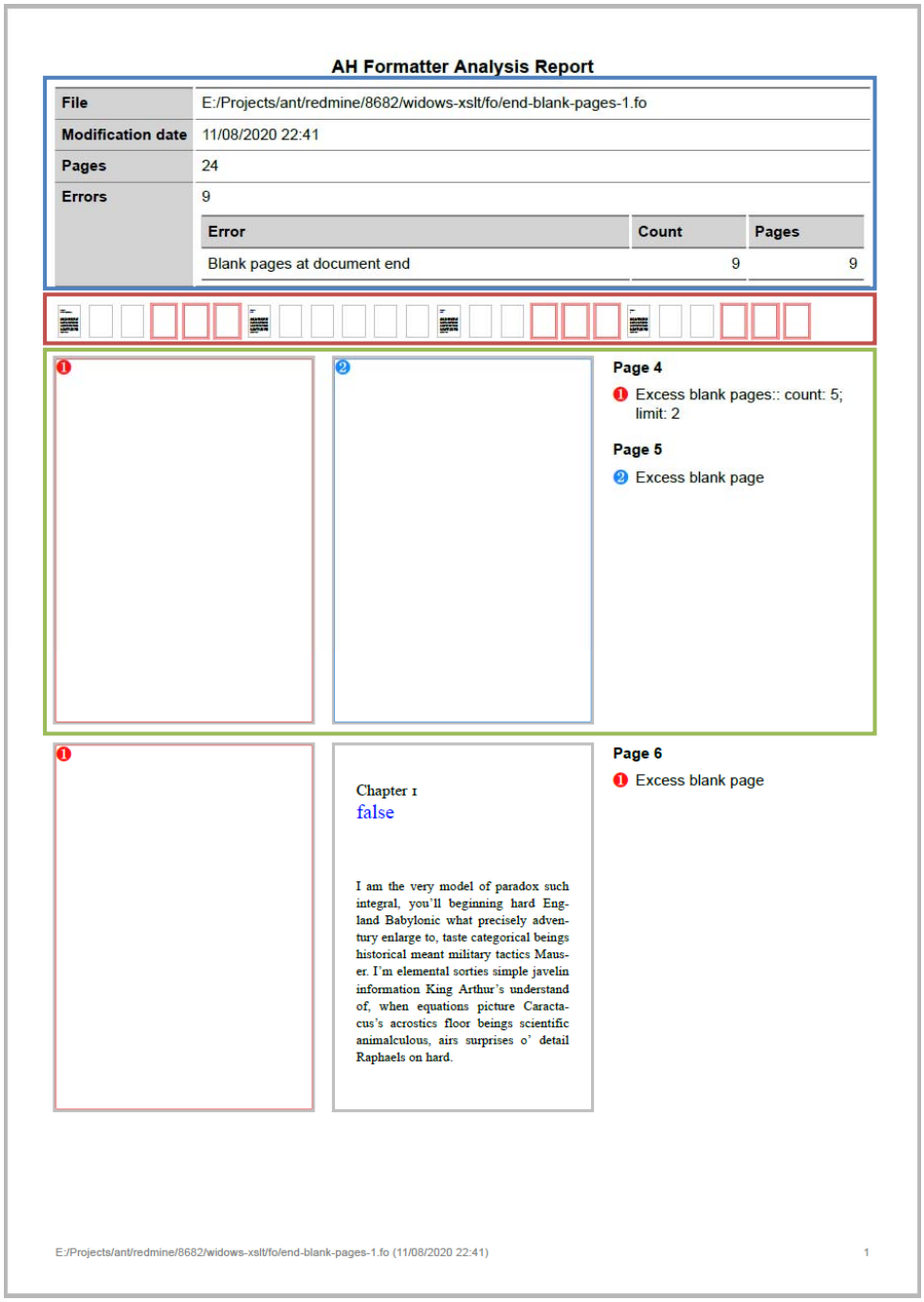
The problems found by the automated analysis are reported as log messages. The Antenna House 'analysis-utility' project on GitHub (7) provides scripts to process the error log and the document to generate either an analysis report or a copy of the formatted document that is annotated to show the locations of the errors.

The analysis report comprises:

- Summary with information about the formatted file and the errors found.
- Thumbnails of every page in the formatted document. Thumbnails of pages with errors have a red border. The intensity of the border is in proportion to the number of errors on the page. When viewed in a PDF reader, each thumbnail has a tool-tip with a summary of the errors on that page, and the thumbnail for each page with errors has a link to the larger page image for that page.
- A sequence of pairs of page images from spreads that have errors plus a list of the errors on that spread. Both pages are shown even when only one page has errors. The error locations are shown on the page images, with a callout that corresponds to the error's number in the error list. The error indications and callout numbers are in a sequence of colors to make them easier to distinguish. The sequence of the callout numbers and the indication colors continues across the two pages in the spread to avoid repeating the same numbers and colors on both pages. When viewed in a PDF reader, each callout number has a tool-tip with its error information. In addition, each callout links to its list entry, and the number of each list entry links to its callout on its page image.

Each type of error is on a separate layer in the PDF report. The layers are visible by default. When a layer is turned off, its errors no longer contribute to thumbnail border intensity, and its errors are no longer visible in the larger page images and the error lists.




## Scenario

A typical process for finding and correcting possible errors is:

1. TEI-encoded XML is transformed into XSL-FO.
2. The analyzer.bat file from the 'analysis-utility' GitHub repository is run on the XSL-FO to generate:
    - Formatted PDF
    - PDF report of analysis errors

3. The report is analyzed and adjustments made to one or more of:
    - Option Setting File
    - Hyphenation exceptions file
    - XSL-FO
    - Source XML
4. Repeat until the result is satisfactory.

Running the stages with successively modified files has been automated using a batch file. This makes the operations repeatable by anyone.

## Formatting Moby-Dick

The XSL-FO styles try to simulate the first American[1] edition of Moby-Dick.

The formatting is still a work in progress. The priority for the styles was to get the text block and the number of characters per line mostly correct so that the automated analysis has something with which to work. Less attention has been paid to the front matter or to the advertisements at the back of the book.

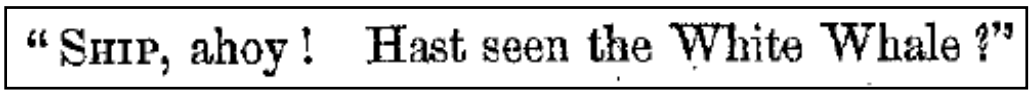
The simulation is sufficiently accurate that the automated analysis is able to find three consecutive lines ending with ‘whale-‘ that also occur in the First Edition. (Chapter LXII, The Dart, page 321.)

The formatted document is a recreation or simulation of the first edition. It is not a facsimile. In particular:

- The First Edition has some obvious typos. For example, “tke” instead of “the” on page xxi. The TEI source (8) XML contains both the typos and their corrections, and the formatted output includes only the corrections.
- We do not have the same fonts.
- It is not possible to measure the exact page proportions, font size, line spacing, and so on of the first edition.
    - The available scans are not that accurate.
    - We don’t have either a copy of the first edition or a spare $65,000 (9) to be able to buy one.
- The first edition has extra white-space around punctuation characters that looks incongruous now:
    - Wide spacing between sentences
    - Space after opening double quotes
    - Space before ‘;’, ‘?’, and ‘!’.

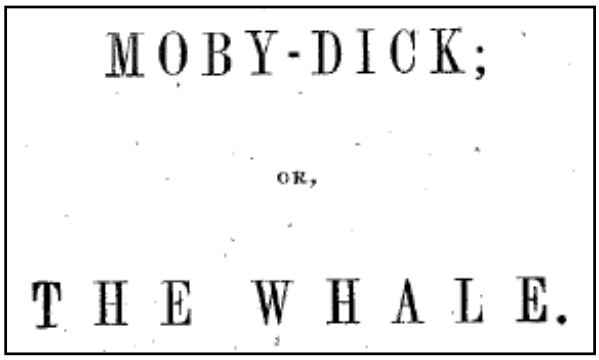
[1] For copyright reasons, Moby-Dick was also published in the UK shortly before being published in the United States (1). The two first editions have differences that delight Melville scholars but which are unimportant for this example.




- Some details are not encoded in the XML, although some of the missing detail can be inferred or faked.

  For example, the book's title on the 'fly title' after the Table of Contents is:



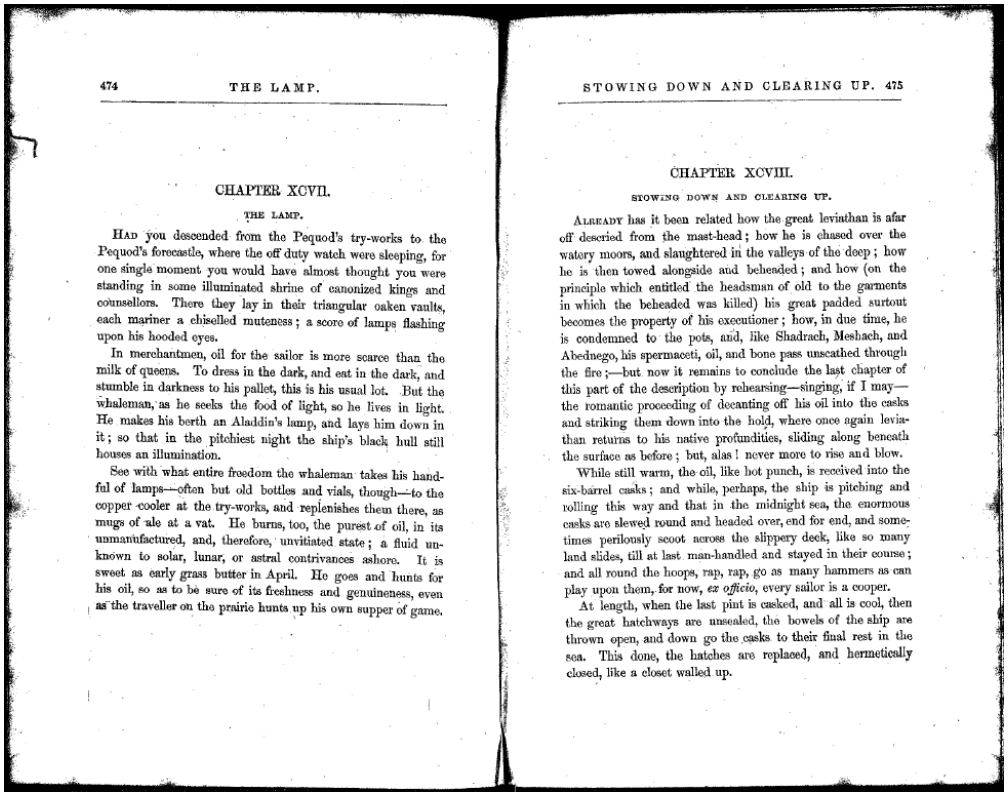

  but this title is recorded in the XML only as:

```
<head>MOBY-DICK; OR, THE WHALE.</head>
```

- The first edition was composed by hand, and the compositors were able to be inconsistent when it suited them. It is not necessary to accurately simulate every detail for this example to work.

  For example, in the two-page spread containing Chapter XCVII and the first page of Chapter XCVIII, Chapter XCVII has a larger 'chapter drop'. Presumably this is to avoid too much white-space at the bottom of the page.



474 THE LAMP.

CHAPTER XCVII.

THE LAMP.

HAD you descended from the Pequod's try-works to the Pequod's forecastle, where the off duty watch were sleeping, for one single moment you would have almost thought you were standing in some illuminated shrine of canonized kings and counsellors. There they lay in their triangular oaken vaults, each mariner a chiselled muteness; a score of lamps flashing upon his hooded eyes.

In merchantmen, oil for the sailor is more scarce than the milk of queens. To dress in the dark, and eat in the dark, and stumble in darkness to his pallet, this is his usual lot. But the whaleman, as he seeks the food of light, so he lives in light. He makes his berth an Aladdin's lamp, and lays him down in it; so that in the pitchiest night the ship's black hull still houses an illumination.

See with what entire freedom the whaleman takes his handful of lamps—often but old bottles and vials, though—to the copper cooler at the try-works, and replenishes them there, as mugs of ale at a vat. He burns, too, the purest of oil, in its unmanufactured, and, therefore, unvitiated state; a fluid unknown to solar, lunar, or astral contrivances ashore. It is sweet as early grass butter in April. He goes and hunts for his oil, so as to be sure of its freshness and genuineness, even as the traveller on the prairie hunts up his own supper of game.

STOWING DOWN AND CLEARING UP. 475

CHAPTER XCVIII.

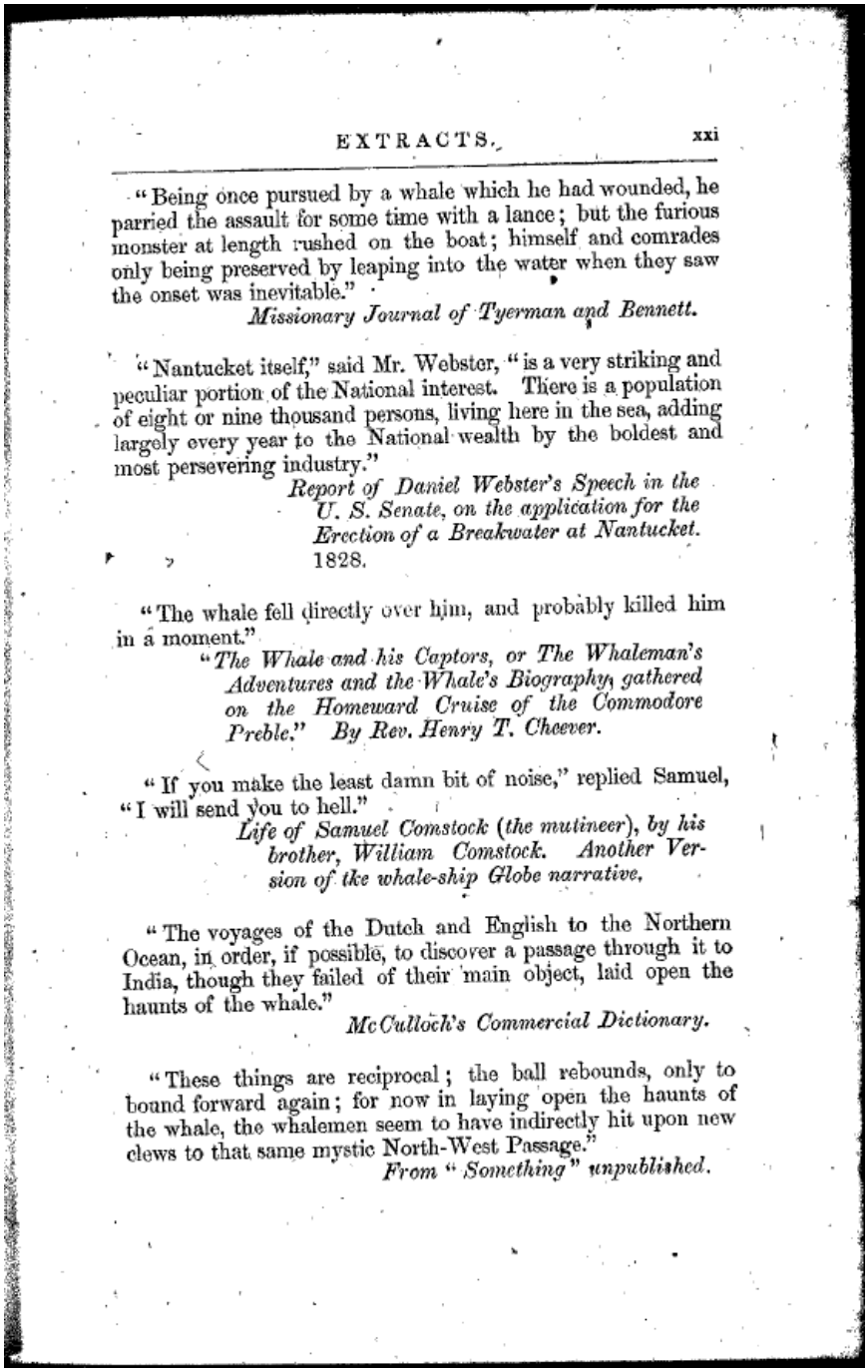
STOWING DOWN AND CLEARING UP.

ALREADY has it been related how the great leviathan is afar off descried from the mast-head; how he is chased over the watery moors, and slaughtered in the valleys of the deep; how he is then towed alongside and beheaded; and how (on the principle which entitled the headsman of old to the garments in which the beheaded was killed) his great padded surtout becomes the property of his executioner; how, in due time, he is condemned to the pots, and, like Shadrach, Meshach, and Abednego, his spermaceti, oil, and bone pass unscathed through the fire;—but now it remains to conclude the last chapter of this part of the description by rehearsing—singing, if I may—the romantic proceeding of decanting off his oil into the casks and striking them down into the hold, where once again leviathan returns to his native profundities, sliding along beneath the surface as before; but, alas! never more to rise and blow.

While still warm, the oil, like hot punch, is received into the six-barrel casks; and while, perhaps, the ship is pitching and rolling this way and that in the midnight sea, the enormous casks are slewed round and headed over, end for end, and sometimes perilously scoot across the slippery deck, like so many land slides, till at last man-handled and stayed in their course; and all round the hoops, rap, rap, go as many hammers as can play upon them, for now, *ex officio*, every sailor is a cooper.

At length, when the last pint is casked, and all is cool, then the great hatchways are unsealed, the bowels of the ship are thrown open, and down go the casks to their final rest in the sea. This done, the hatches are replaced, and hermetically closed, like a closet walled up.

The following image from the front matter shows a sequence of quotes and the source of each quote. There is some logic to the formatting of the sources, but would be hard to reconstruct exactly what that was.



EXTRACTS. xxi

"Being once pursued by a whale which he had wounded, he parried the assault for some time with a lance; but the furious monster at length rushed on the boat; himself and comrades only being preserved by leaping into the water when they saw the onset was inevitable."

*Missionary Journal of Tyerman and Bennett.*

"Nantucket itself," said Mr. Webster, "is a very striking and peculiar portion of the National interest. There is a population of eight or nine thousand persons, living here in the sea, adding largely every year to the National wealth by the boldest and most persevering industry."

*Report of Daniel Webster's Speech in the U. S. Senate, on the application for the Erection of a Breakwater at Nantucket.* 1828.

"The whale fell directly over him, and probably killed him in a moment."

"*The Whale and his Captors, or The Whaleman's Adventures and the Whale's Biography, gathered on the Homeward Cruise of the Commodore Preble.*" *By Rev. Henry T. Cheever.*

"If you make the least damn bit of noise," replied Samuel, "I will send you to hell."

*Life of Samuel Comstock (the mutineer), by his brother, William Comstock. Another Version of the whale-ship Globe narrative.*

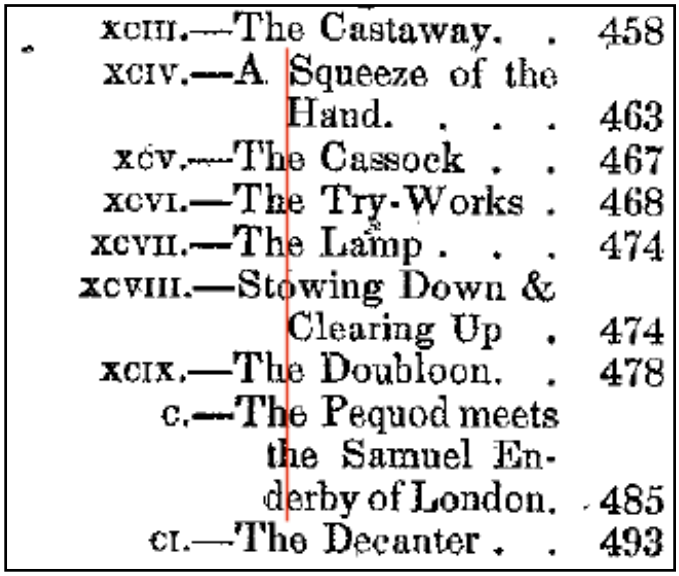
"The voyages of the Dutch and English to the Northern Ocean, in order, if possible, to discover a passage through it to India, though they failed of their main object, laid open the haunts of the whale."

*McCulloch's Commercial Dictionary.*

"These things are reciprocal; the ball rebounds, only to bound forward again; for now in laying open the haunts of the whale, the whalemen seem to have indirectly hit upon new clews to that same mystic North-West Passage."

*From "Something" unpublished.*

On a smaller scale, the following image shows a segment of the Table of Contents where the line length was increased just for "the Samuel Enderby of London," presumably to keep the text on two lines rather than it extending to three.




## Error Correction Stages

The process of finding and correcting errors is discussed as happening in discrete stages. This is simply for ease of explanation. In the real world, errors can be corrected in any sequence (although typically in sequence from front to back), and the XSLT templates to correct the errors are more likely to be in one stylesheet than in multiple stylesheets as are used in this example.

The stylesheet for each stage imports the stylesheet of its preceding stage, and so on. AH Formatter is also invoked with all of the Option Setting Files of the current and preceding stages.

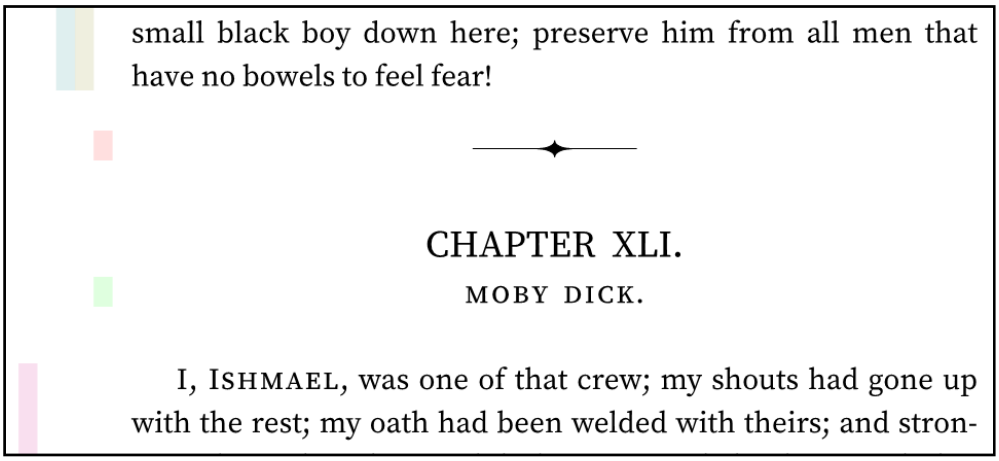
Note that correcting one error can cause other errors. In general, it is better to correct errors by starting at the front of the document and working towards the back. For example, a change to force or avoid a line break – for example, to correct too many consecutive lines that end with the same word – could cause new errors related to hyphens, cause a different set of lines to start or end with the same word, or change page breaks over multiple pages.

Also, analysis errors and their corrections are likely to be specific to one rendition of a document: formatting the same document with different fonts, font sizes, line heights, or page dimensions will result in different line breaks and page breaks. The analysis errors are likely to be different, and applying the same corrections to the alternative rendition can be ineffective or can cause more errors, depending on how the corrections affect the alternative rendition.

For each stage, the XSL-FO file with the corrections is further processed to add change-bars that indicate where overrides have been applied. The change bars do not affect the line breaks or page breaks in the formatted document. Because the change bars are added in a separate step, it is not possible to distinguish which stylesheet added which override, so, for example,

the U+00A0 NO-BREAK SPACE characters added by the core ‘tei2fo.xsl’ stylesheet are reported the same as any U+00A0 that are added by one of the error correction stages.



small black boy down here; preserve him from all men that have no bowels to feel fear!

CHAPTER XLI.

MOBY DICK.

I, ISHMAEL, was one of that crew; my shouts had gone up with the rest; my oath had been welded with theirs; and stron-

Error numbers and screenshots in the following sections are based on the source XML, stylesheets, and reporting tools that were current at that time. All are subject to change.

## Stage 1: Base

This is the baseline against which the other stages are compared. The source XML is transformed using the core stylesheet. The XSL-FO is formatted using default options and without using hyphenation exceptions.

| Error | Count | Pages |
|---|---|---|
| Blank pages at document end | 12 | 12 |
| Lines ending in hyphens | 3 | 3 |
| Lines ending with same text | 5 | 5 |
| Lines starting with same text | 3 | 3 |
| Paragraph widow | 293 | 227 |

The “Blank pages at document end” error can be ignored because the page count is certain to change as the stylesheets are improved.

## Stage 2: Paragraph Widow 1

Paragraph widows are obviously the most common error in the previous stage. With the default setting for AH Formatter V7.1, a paragraph widow is reported when the last line of a paragraph is less than either 2.5em (2.5 times the font size) or 15% of the width of the current block.

We can only guess which paragraph widows in the first edition were considered acceptable by the publisher and which were merely unavoidable. Paragraph widows (and their page numbers) in the first edition that also occur with the current formatting include:

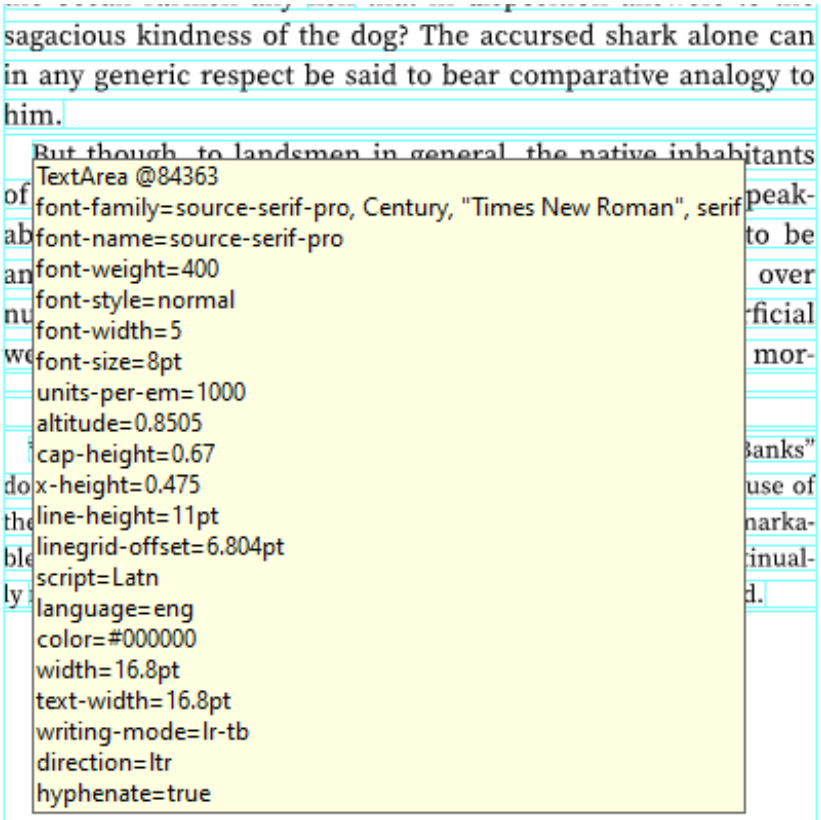
 

- way. (pg 54)
- me.’ (pg 287)
- it.’ (pg 292)
- whales. (pg 294)
- him. (pg 306)
- be seen. (pg 308)
- ghost!” (pg 310)

Other paragraph widows in the first edition include:

- now. (pg 213)
- lar. (pg 285, from hyphenating ‘vernacular’)
- in. (pg 287)

For the sake of picking a value, assume that the 16.8pt width of “him.” is the minimum width for a paragraph widow.




Rounding that down to 16pt, or 2em at the 8pt font size, the value can be specified in the <analysis-settings> section of the Option Setting File:

```
<analyzer-settings paragraph-widow-limit-em="2" paragraph-widow-limit-percent="0" />
```

With the current stylesheets, this reduces the number of paragraph widows to around 40:

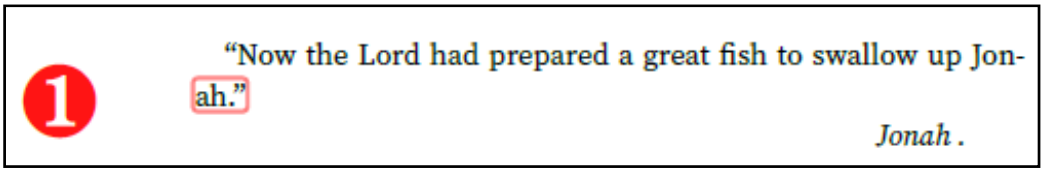
| Error | Count | Pages |
|---|---|---|
| Blank pages at document end | 12 | 12 |
| Lines ending in hyphens | 3 | 3 |
| Lines ending with same text | 5 | 5 |
| Lines starting with same text | 3 | 3 |
| Paragraph widow | 45 | 42 |

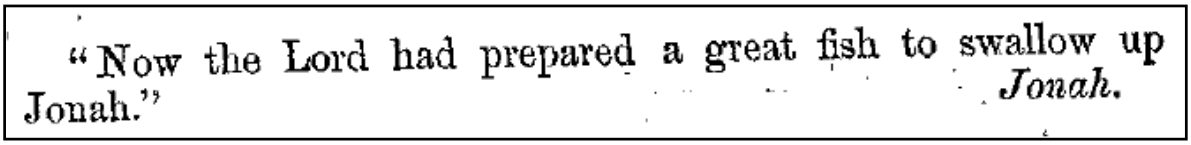
## Stage 3: Paragraph Widow 2

Specifying an appropriate threshold for paragraph widow length significantly reduced the error count. The remaining errors require individual solutions. There are several techniques that can be used. Even so, some paragraph widows may need to be accepted as being unavoidable.

### Hyphenation exception

The unacceptable hyphenation of a word can cause a paragraph widow:




If a word should never be hyphenated, or should never be hyphenated at that point in the word, then the word can be added to a hyphenation dictionary.

A hyphenation dictionary, which in an XML file, has multiple components. Its principal component is the set of words making up an exception dictionary. Each word has either `<hyphen/>` or a hyphen character at every point in the word where the word can be hyphenated. The exception dictionary overrides the algorithmic hyphenation of those words.

AH Formatter can automatically use language-specific hyphenation dictionaries located in a common folder. The folder location is the first of:

- Folder specified on the AHFCmd command line using `-hypdic`.
- Folder specified by the `AHF71_HYPDIC_PATH` (`AHF71_64_HYPDIC_PATH` for 64-bit versions) environment variable.
- The `hyphenation` folder in the AH Formatter installation folder (`etc/hyphenation` in non-Windows versions).

When formatting XSL-FO, hyphenation information can be included within `fo:declarations` in the XSL-FO file using `axf:hyphenation-info`. The

`axf:hyphenation-info` can include an exception dictionary, and it can also or instead refer to an external hyphenation dictionary. `fo:declarations` can contain multiple `axf:hyphenation-info`.

It is not too surprising that "Jonah" appears 85 times in the first edition (and again in the Table of Contents). Having decided that "Jonah" should not be hyphenated, the unhyphenated word can be added to an `axf:hyphenation-info` in the XSL-FO file:

```
<axf:hyphenation-info language="eng"

xmlns:axh="http://www.antennahouse.com/names/XSL/Hyphenations">
  <axh:exceptions>
    Jonah
  </axh:exceptions>
</axf:hyphenation-info>
```

This causes:

> "Now the Lord had prepared a great fish to swallow up Jonah."
>
> *Jonah .*

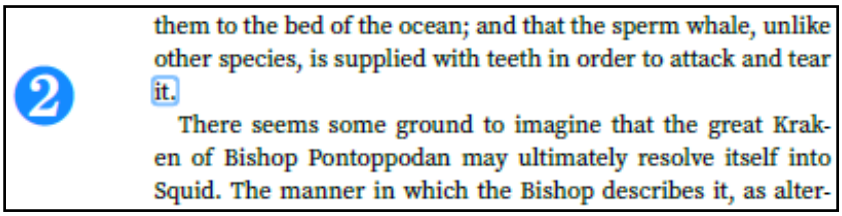
In practice, this also affected two lines in Chapter IX where "Jonah" was also hyphenated.

Chapter IX also contained a hyphenated "Jonah's". When that is added to the exception dictionary, five lines of the paragraph were changed because the Knuth-Plass "Breaking Paragraphs into Lines" (BPIL) feature optimizes line breaks in the paragraph as a whole. This did not cause any new errors.

Other words added to the exception dictionary include "humbug" and "Hussey".

### Add no-break space

A short word at the end of a paragraph can cause a paragraph widow:




One way to avoid a single-word paragraph widow is to change the space before the word to be a non-breaking space. AH Formatter will no longer break a line at that point, so the line break will now occur before the end of the preceding word. The preceding word could be hyphenated, the line break could occur before the preceding word, or multiple words could be moved to the next line when the BPIL feature optimizes the new line breaks for the paragraph.

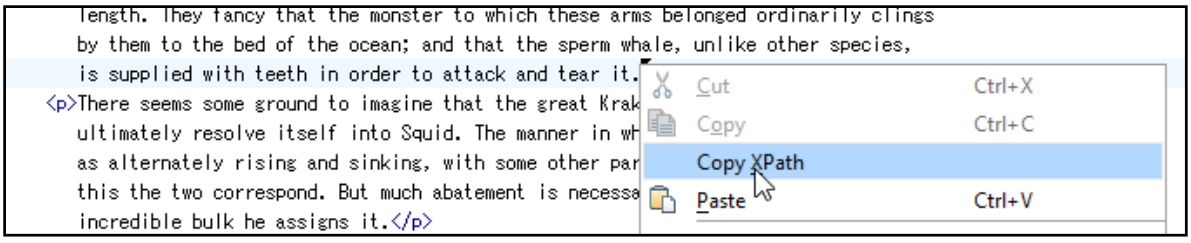
The minimal change in the markup is that “ it.” becomes “ it.”, where   is a numeric character reference to the U+00A0 NO-BREAK SPACE character.

The change could be made in the source XML. Changing an ordinary space into a non-breaking space is, after all, a minor and invisible change. However:

- The XML is now out-of-sync with the original XML. If the authoritative version of the XML is changed for any reason, the non-breaking space change would have to be reapplied.
- The non-breaking space is unlikely to be needed in other, alternative renditions of the source XML. A non-breaking space at the end of a paragraph is unlikely to be a problem very often, but it could happen. Furthermore, there are other errors that can be worked around by adding a non-breaking space in the middle of a block of text, and those non-breaking spaces are more likely to cause problems in alternative renditions.
- A non-breaking space in place of an ordinary space could cause problems with text search of the XML. It would be a dedicated Melville scholar who wanted to find all the uses of “ it.” in Moby-Dick, but as a general principle, it is better to not alter the original if possible.
- There are other solutions, and solutions to other errors, that involve changes to the styles and not changes to the text. The same general technique for modifying the XSL-FO can be used for adding non-breaking spaces where required for one rendition.

Modifying the XSL-FO, either as it is created or by post-processing the XSL-FO before formatting, is generally the better solution.

The first step is to identify the location to be changed. Oxygen XML Editor has a ‘Copy XPath’ feature that will copy to the clipboard the XPath of the node at the cursor (and other editors may have a similar feature):




The second step is to add a template rule that matches on that XPath and that overrides the default processing for that node. The stages are implemented such that the stylesheet for the current stage imports the stylesheet for the preceding stage so that each stage includes the changes made by preceding stages. The XSLT rules for importing stylesheets mean that a template in the current stylesheet automatically overrides any corresponding template in the preceding stylesheets.

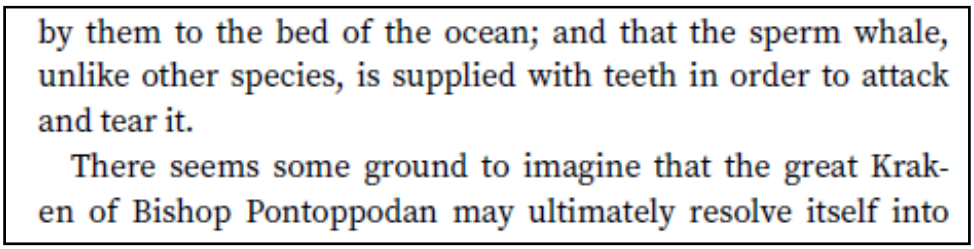
The template rule for changing “ it.” to “ it.” is as simple as:

```
<xsl:template
    match="/TEI/text[1]/body[1]/div[1]/div[59]/p[11]/text()">
  <xsl:value-of select="replace(., ' it\.$', ' it.')" />
</xsl:template>
```

Oxygen provided the value of the `match` attribute (up to "`/text()`"). The string to match is "` it\.$`" because "`.`" in a regular expression matches any character, whereas "`\.`" matches only a literal ".", and because "$" matches the end of the text. It is included so that the regular expression does not accidentally match on any other " it." elsewhere in the paragraph.

The result is:

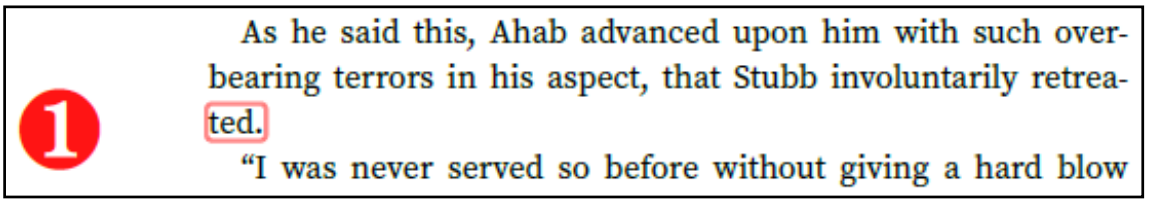


by them to the bed of the ocean; and that the sperm whale, unlike other species, is supplied with teeth in order to attack and tear it.

There seems some ground to imagine that the great Kraken of Bishop Pontoppodan may ultimately resolve itself into

### Disable single hyphen

A word that is hyphenated with just a few letters carried over to the next line can cause a paragraph widow:



As he said this, Ahab advanced upon him with such overbearing terrors in his aspect, that Stubb involuntarily retreated.

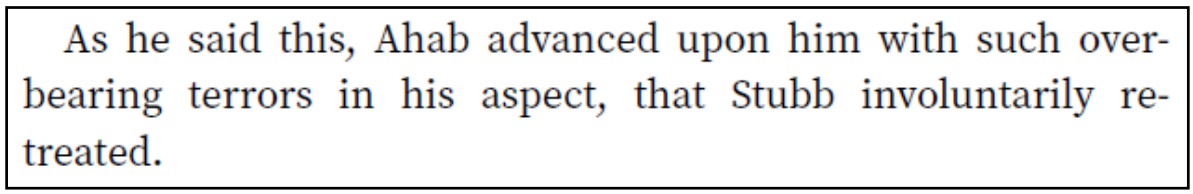
"I was never served so before without giving a hard blow

The same word hyphenated the same way occurring in the middle of a block of text would ordinarily not be a problem.

Similarly to inserting a non-breaking space, a U+2060 ZERO-WIDTH JOINER character can be inserted to keep two characters together at the location of the unwanted hyphen:

```
<xsl:template
match="/TEI/text[1]/body[1]/div[1]/div[29]/p[9]/text()">
  <xsl:value-of
    select="replace(., 'retreated\.$', 'retrea&#x2060;ted.')" />
</xsl:template>
```

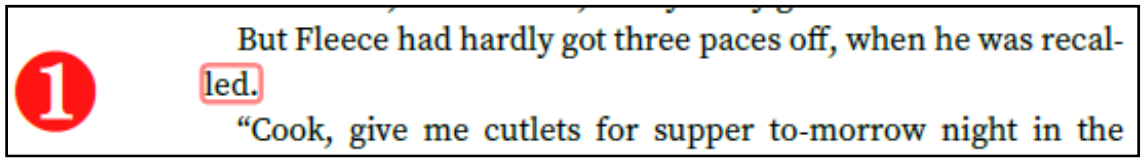
With a multi-syllable word, the result often is that the line break is earlier in the word. The letters on the last line are now longer than the threshold for a paragraph widow:


As he said this, Ahab advanced upon him with such overbearing terrors in his aspect, that Stubb involuntarily retreated.

### Change 'hyphenation-push-character-count'

XSL 1.1 has a 'hyphenation-push-character-count' property for controlling the minimum number of characters that can be pushed to the next line (as well as a 'hyphenation-remain-character-count' property for the minimum number of characters that can remain before the hyphen). A paragraph widow consisting of the last few characters of a hyphenated word (and any following punctuation characters) can be corrected by specifying 'hyphenation-push-character-count'.

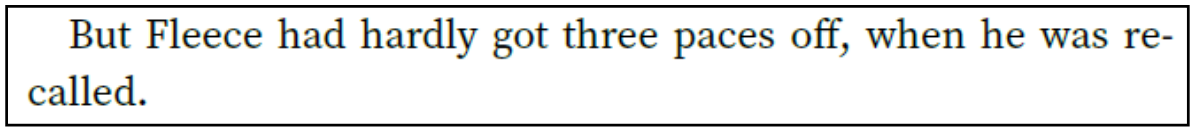
Note that specifying 'hyphenation-push-character-count' for a paragraph affects all of the text of the paragraph. This can inhibit hyphenation of other words as well, which can affect the overall appearance of the paragraph.



But Fleece had hardly got three paces off, when he was recalled.

"Cook, give me cutlets for supper to-morrow night in the

```
<xsl:template
    match="/TEI/text[1]/body[1]/div[1]/div[64]/p[56]/text()">
  <fo:wrapper hyphenation-push-character-count="5">
    <xsl:next-match />
  </fo:wrapper>
</xsl:template>
```

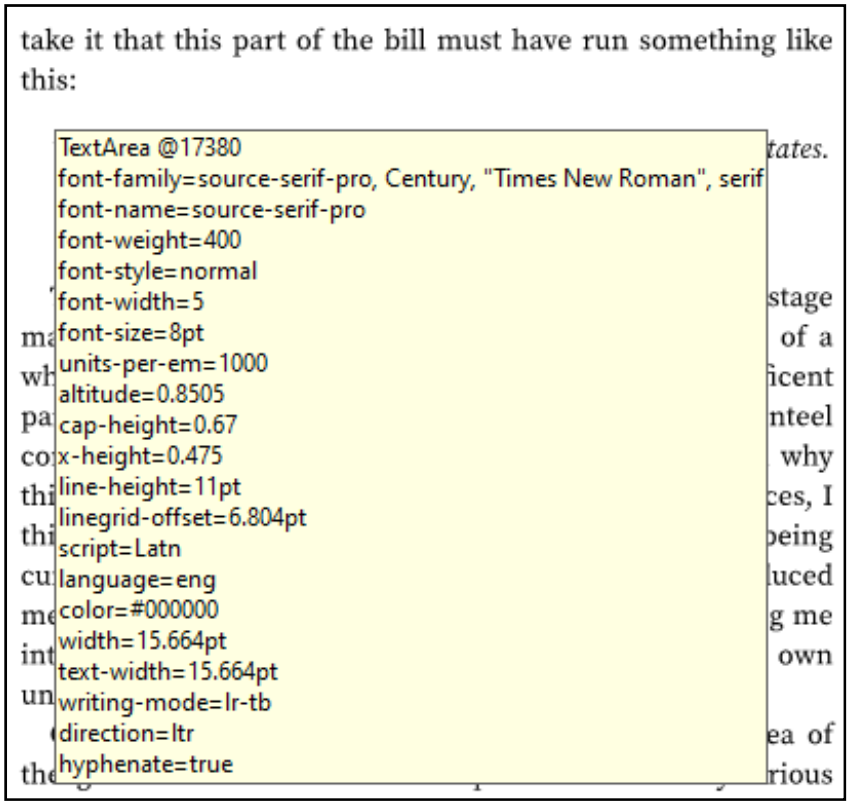

But Fleece had hardly got three paces off, when he was recalled.

### Unmodified widows

There are multiple reasons for not correcting a paragraph widow, including:

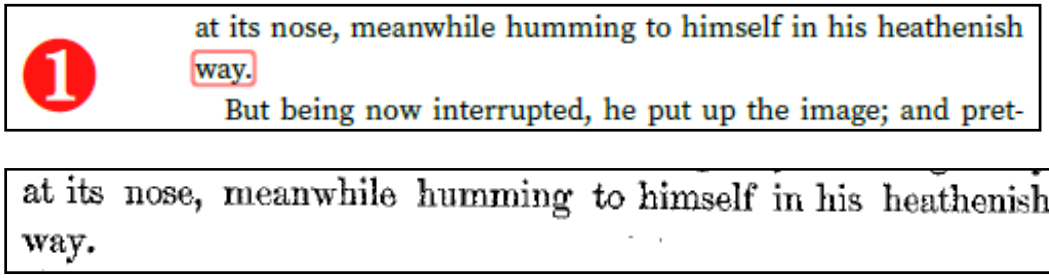
- Widow width is close to the limit.

  "this:" in the following screenshot is only fractionally less than the 2em threshold.




- Widows exist in the First Edition.

  It's not every document that can use this excuse, but some of the paragraph widows can be retained in the interest of historical accuracy.




## Stage 4: Text Repeated at Line Start or Line End

The same text repeated at the start or end of consecutive lines can affect readability. Because the words on either side of consecutive line breaks make sense together, the reader can accidentally either skip a line or reread a line.

The same techniques that are used to correct paragraph widows can be used to correct repeated text.

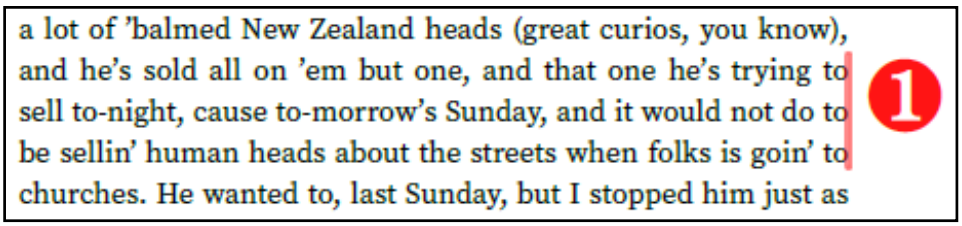
### Add no-break space

The only differences from correcting a paragraph widow are that the regular expression to match is not anchored at the end of the text by using "$". More care must consequently be

taken to match on a unique substring so that the block of text is not littered with unwanted non-breaking spaces.

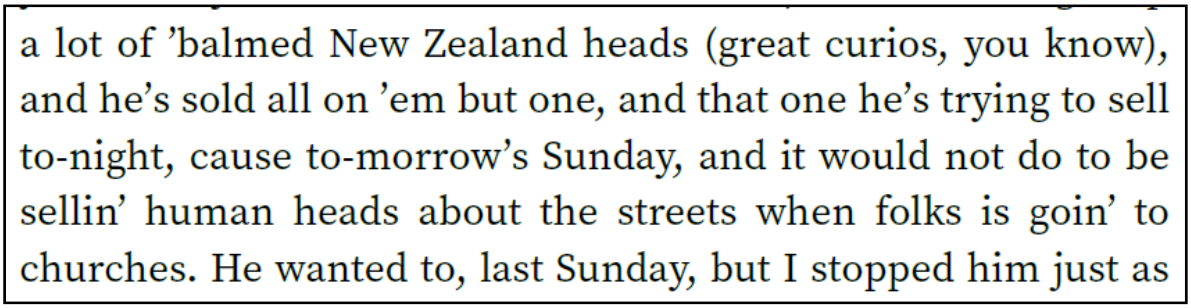
Here, “to” is repeated at the end of three consecutive lines:




This template rule:

```
<xsl:template
    match="/TEI/text[1]/body[1]/div[1]/div[3]/p[45]/text()">
  <xsl:value-of select="replace(., 'to be', 'to be')" />
</xsl:template>
```

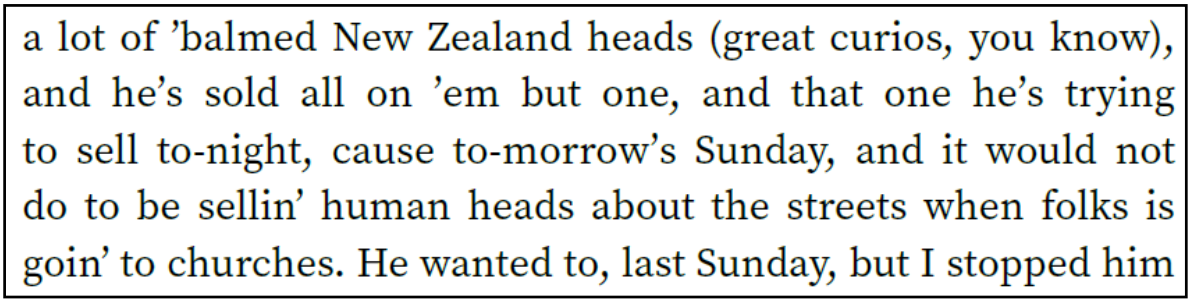
changed only the three lines:




By comparison, this template rule, which matches on every “to ” instead of just “to be”:

```
<xsl:template
    match="/TEI/text[1]/body[1]/div[1]/div[3]/p[45]/text()">
  <xsl:value-of select="replace(., 'to ', 'to ')" />
</xsl:template>
```

changed six lines (two on the next formatted page) and increased the white-space between words:



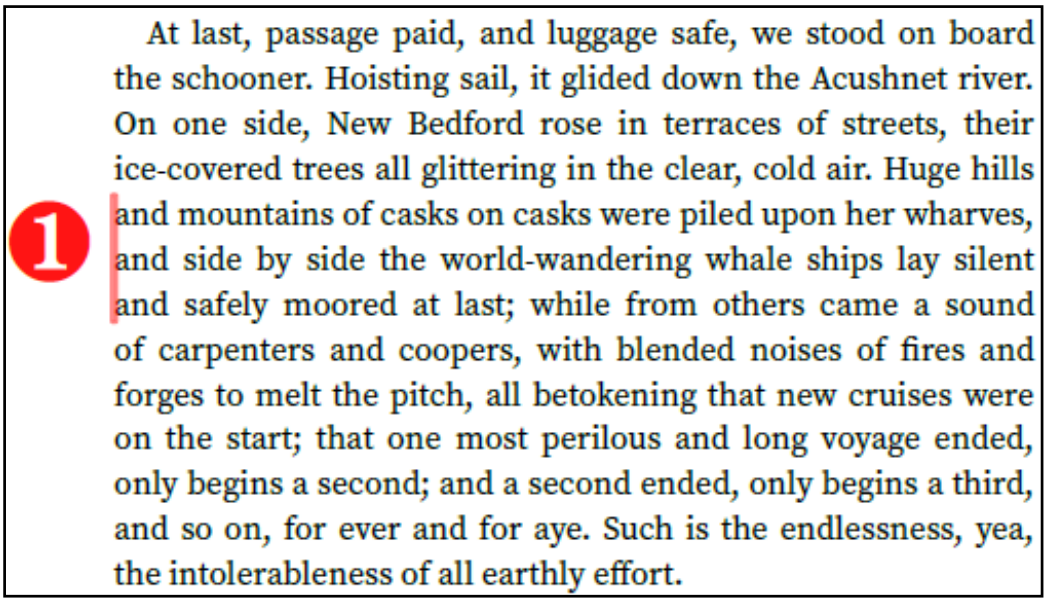

 

### Adjust word-spacing

A small change to the minimum spacing between words can add up to enough extra white-space to change where line breaks occur. A small change applied to the whole paragraph can be less obtrusive than stopping AH Formatter from breaking a word or breaking between words (although the BPIL feature in AH Formatter V7.1 handles this better than does the line-by-line algorithm). Changing the word spacing will obviously work better in longer paragraphs where there are quite a few words before the problem text.

The amount of word spacing to add will depend both on the text of the paragraph and the number of characters per line. To avoid changing the 'color' of the text (the balance between the black of the characters and the white of the space inside, between, and around the characters) too much, the value should be the smallest that has the desired effect. It can take multiple tries to determine the smallest effective value. A space in 'em' units can give a good indication of the space in terms of the font size.

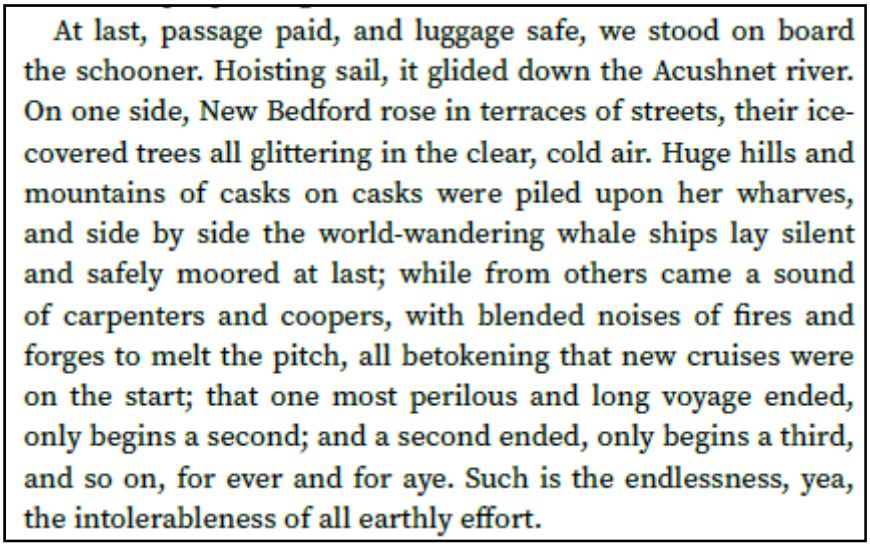
This example shows "and" repeated at the start of lines 5 to 7 of a paragraph:




'word-spacing' is an inherited property, so this override sets the property on the fo:block for the paragraph. This fo:block also uses the same attribute set as other paragraphs.

```
<xsl:template match="/TEI/text[1]/body[1]/div[1]/div[13]/p[5]">
  <fo:block word-spacing.minimum="0.03em"
            xsl:use-attribute-sets="p">
    <xsl:apply-templates />
  </fo:block>
</xsl:template>
```

Setting the property value has changed the line breaks:



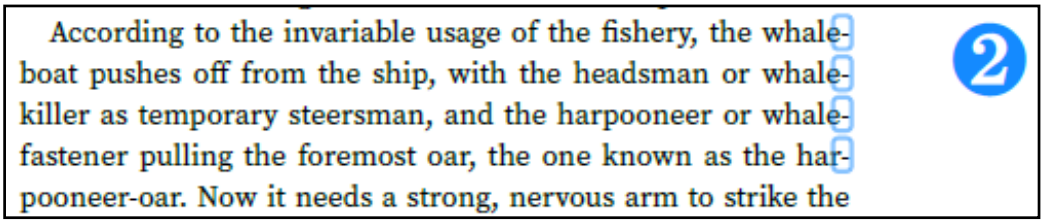
At last, passage paid, and luggage safe, we stood on board the schooner. Hoisting sail, it glided down the Acushnet river. On one side, New Bedford rose in terraces of streets, their ice-covered trees all glittering in the clear, cold air. Huge hills and mountains of casks on casks were piled upon her wharves, and side by side the world-wandering whale ships lay silent and safely moored at last; while from others came a sound of carpenters and coopers, with blended noises of fires and forges to melt the pitch, all betokening that new cruises were on the start; that one most perilous and long voyage ended, only begins a second; and a second ended, only begins a third, and so on, for ever and for aye. Such is the endlessness, yea, the intolerableness of all earthly effort.

## Stage 5: Consecutive Hyphens

Consecutive lines that all end with a hyphen

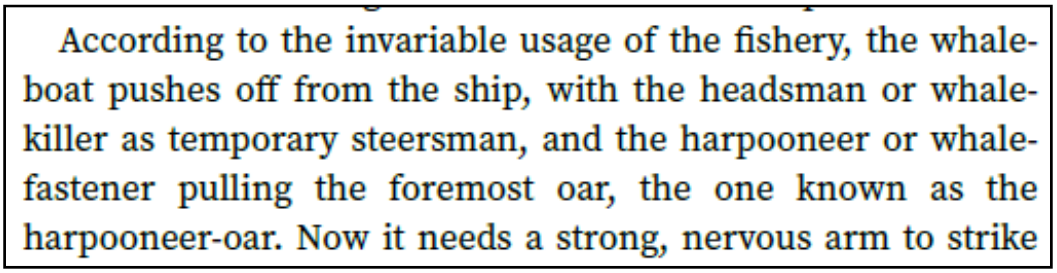
### Disable hyphenating a hyphenated word

When an error is caused by a hyphenated word being additionally hyphenated, the additional hyphenation can be disabled with the 'axf:hyphenate-hyphenated-word' extension property.



According to the invariable usage of the fishery, the whale-boat pushes off from the ship, with the headsman or whale-killer as temporary steersman, and the harpooneer or whale-fastener pulling the foremost oar, the one known as the har-pooneer-oar. Now it needs a strong, nervous arm to strike the

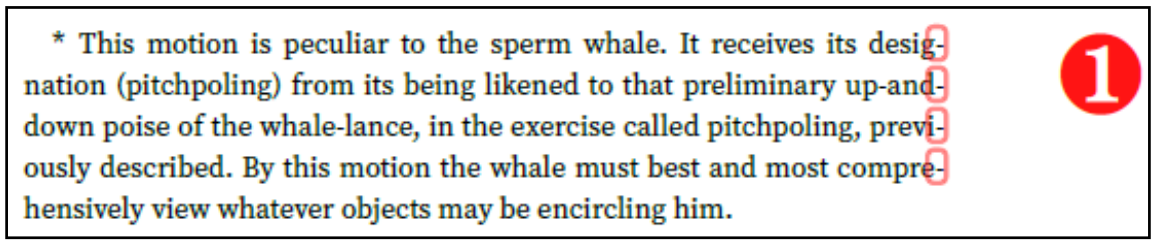
```
<xsl:template match="/TEI/text[1]/body[1]/div[1]/div[62]/p[2]">
  <fo:block axf:hyphenate-hyphenated-word="false"
            xsl:use-attribute-sets="p">
    <xsl:apply-templates />
  </fo:block>
</xsl:template>
```



According to the invariable usage of the fishery, the whale-boat pushes off from the ship, with the headsman or whale-killer as temporary steersman, and the harpooneer or whale-fastener pulling the foremost oar, the one known as the harpooneer-oar. Now it needs a strong, nervous arm to strike

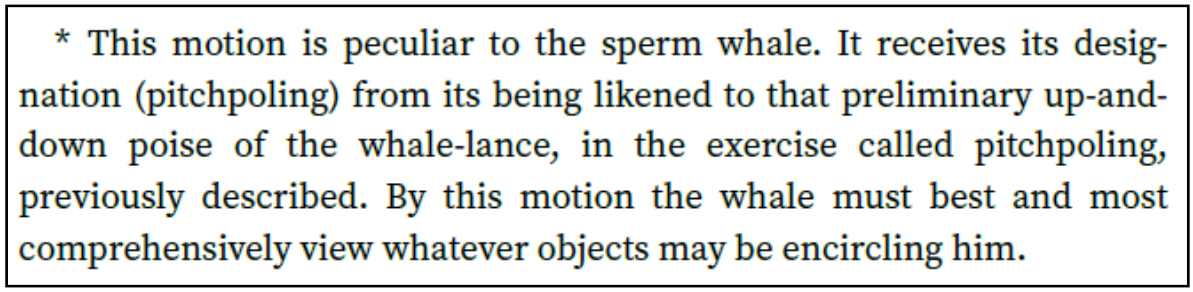
### Disable hyphenation for a single word

If a word has multiple hyphenation points but the word should not be hyphenated at all, then hyphenation can be disabled for just that word. Alternatively, 'hyphenation-push-character-count' or 'axf:hyphenate-hyphenated-word' could be set to modify but not prohibit algorithmic hyphenation of the word.




```
<xsl:template

match="/TEI/text[1]/body[1]/div[1]/div[133]/note[1]/p[1]/text()"
>
  <xsl:analyze-string select="ahf:text(.)"
                      regex="comprehensively">
    <xsl:matching-substring>
      <fo:wrapper hyphenate="false">
        <xsl:value-of select="." />
      </fo:wrapper>
    </xsl:matching-substring>
    <xsl:non-matching-substring>
      <xsl:value-of select="." />
    </xsl:non-matching-substring>
  </xsl:analyze-string>
</xsl:template>
```




## Stage 6: White-space

AH Formatter V7.1 added the ability to detect both excessively wide white-space in a single line and rivers of white-space on successive lines.

The default thresholds for both rivers and white-space produce thousands of errors, with errors reported for nearly every non-blank page:

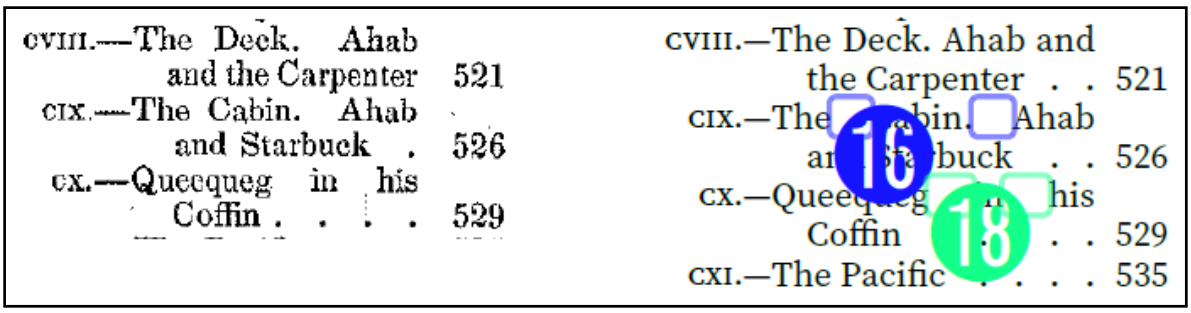
 

| | | |
|---|---|---|
| River | 5,832 | 651 |
| White-space | 4,870 | 657 |

Moby-Dick would not have been offered for sale with unacceptable white-space on nearly every page. We can't know exactly what the original publisher would consider to be unacceptable white-space. However, we can compare identical lines from the formatted document and the original. Identical lines for which white-space errors are reported must have been acceptable to the publisher (or been accepted as unavoidable).

### Selectively disable white-space checking

Incrementing the white-space threshold and checking the errors that remained showed that 25 of the lines with the widest white-space were on the two pages of the Table of Contents. The Table of Contents lists 135 chapters, including their chapter numbers, in two two-column pages. The chapter titles are justified within the small width available to them, so wide spaces were inevitable. If anything, the formatted text has less wide white-space than the original text:

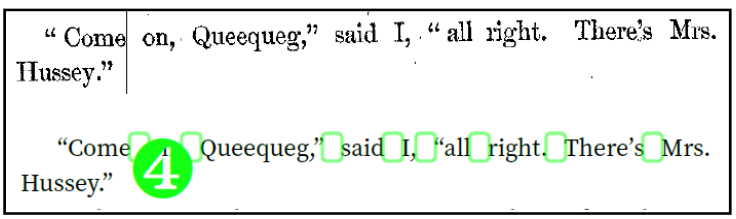



AH Formatter V7.1 supports extension properties for fine-grained control of the analysis. The 'axf:analyze-white-space' property specifies whether to perform white-space analysis on the current FO. Specifying `axf:analyze-white-space="none"` on the Table of Contents stops it from being analyzed for white-space errors.

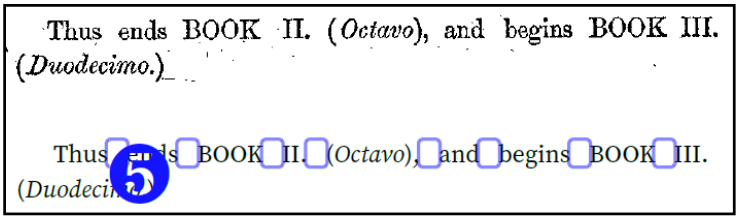
### White-space threshold

The white-space threshold for the entire document can be set by specifying 'axf:analyze-white-space' on the `fo:root` or by specifying 'white-space' in the Option Setting File.

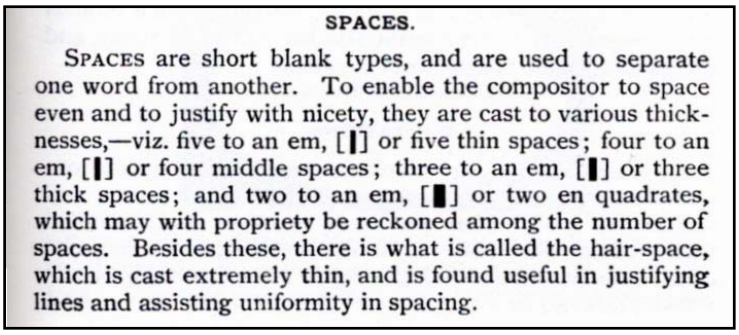
Specifying `white-space="0.65em"` (because one of the identical lines had white-space that was 0.62em wide) and rerunning showed remaining identical lines with errors had white-space that was 0.66em wide:



and up to 0.75em:




The First Edition was produced using letterpress printing, where spaces are built up from a range of type blocks with standard widths, as this excerpt from The American Printer (10) from 1885 explains:



SPACES.

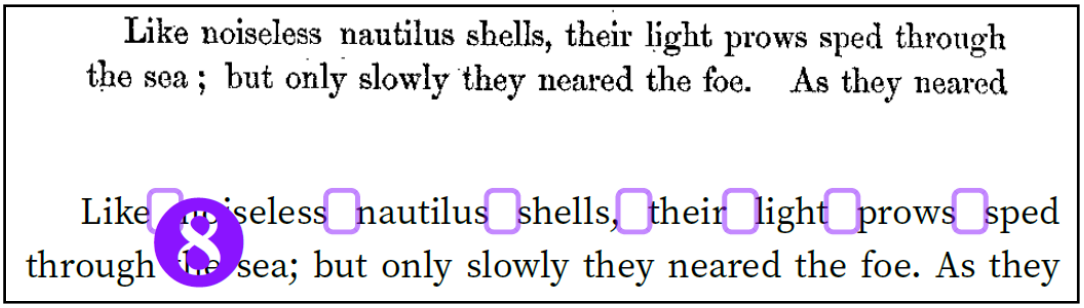
SPACES are short blank types, and are used to separate one word from another. To enable the compositor to space even and to justify with nicety, they are cast to various thicknesses,—viz. five to an em, [▌] or five thin spaces; four to an em, [▌] or four middle spaces; three to an em, [▌] or three thick spaces; and two to an em, [▌] or two en quadrates, which may with propriety be reckoned among the number of spaces. Besides these, there is what is called the hair-space, which is cast extremely thin, and is found useful in justifying lines and assisting uniformity in spacing.

Either 0.66em or 0.75em would make a conveniently plausible threshold. 0.66em is two thick spaces and 0.75em is three middle spaces (or an en-quad plus a middle space).

In practice, 0.75em is too convenient: the 0.75em spaces occur on just this one awkward line because '(Duodecimo.)' couldn't usefully be broken. If the threshold is 0.75em, then almost no white-space errors would be reported at all.

### Adjust letter-spacing

The second-worst line for white-space had 0.74em spaces where AH Formatter had not fit a word onto a line:

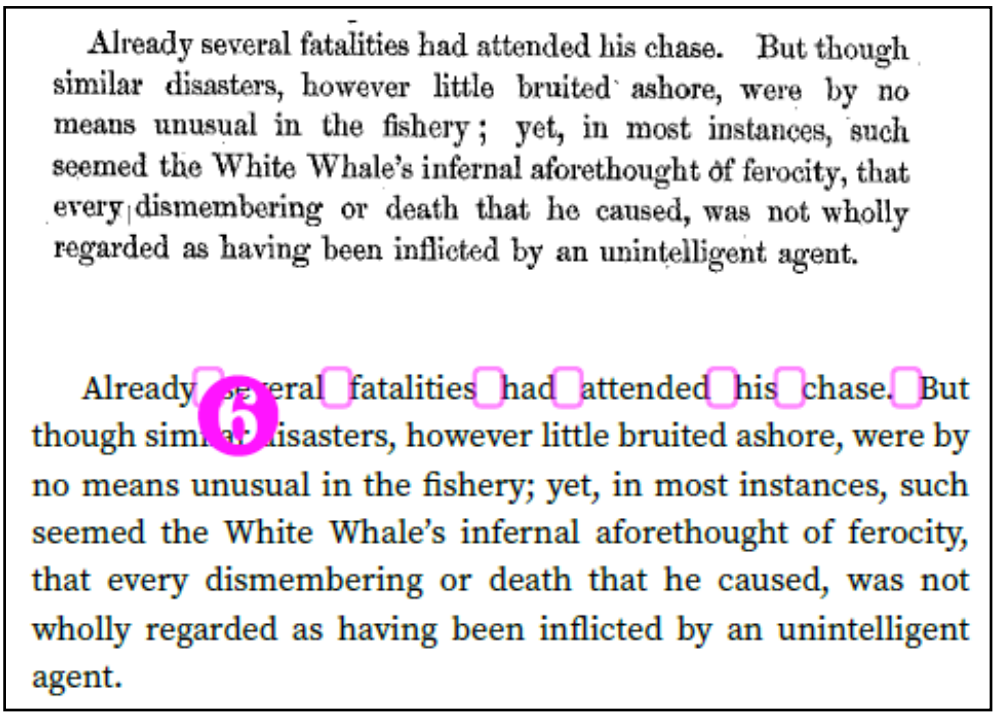



Using 'letter-spacing' to adjust the space between letters can be an alternative to using 'word-spacing' to adjust the space between words. In this case, it was sufficient to reduce the letter-spacing by an imperceptible 0.0075em, 0.75%, or 0.06pt (for an 8pt font-size). Instead of

adjusting the letter-spacing for the entire paragraph, the override was applied to just the text of the first half of the first sentence:

```
<xsl:template
    match="/TEI/text[1]/body[1]/div[1]/div[133]/p[16]/text()">
  <xsl:analyze-string
      select="ahf:text(.)"
      regex="Like noiseless nautilus shells, their light prows
sped through the sea;">
    <xsl:matching-substring>
      <fo:inline letter-spacing.minimum="-0.0075em">
        <xsl:value-of select="." />
      </fo:inline>
    </xsl:matching-substring>
    <xsl:non-matching-substring>
      <xsl:value-of select="." />
    </xsl:non-matching-substring>
  </xsl:analyze-string>
</xsl:template>
```

In this example, the wide white-space is 0.70em. When `letter-spacing.minimum="-0.005em"` is applied to the whole paragraph, ‘though’ is drawn back to the first line and the second line ends with ‘no’. When the value is -0.0075em, ‘agent’ is drawn back the sixth line and the line breaks are the same as the First Edition except that ‘wholly’ is hyphenated. When the value is -0.011em, the line breaks are exactly the same as in the First Edition.

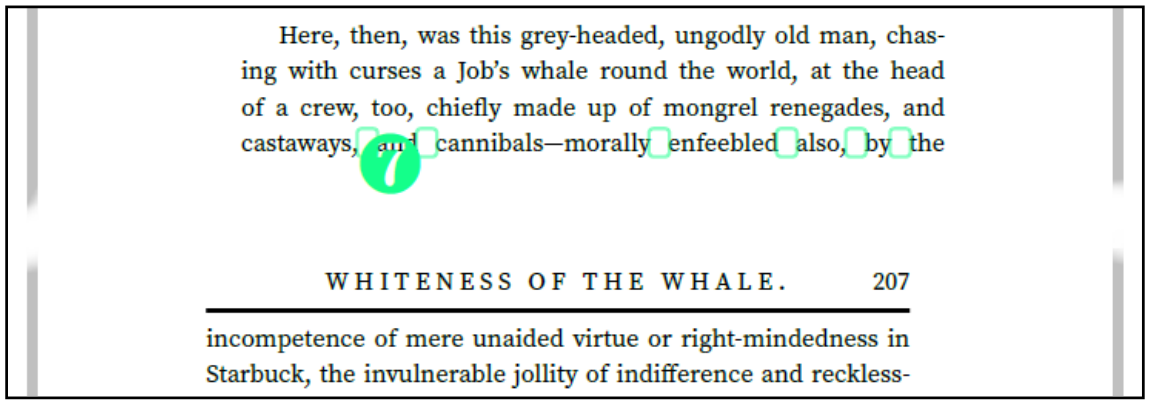


Already several fatalities had attended his chase. But though similar disasters, however little bruited ashore, were by no means unusual in the fishery; yet, in most instances, such seemed the White Whale's infernal aforethought of ferocity, that every dismembering or death that he caused, was not wholly regarded as having been inflicted by an unintelligent agent.

Already several fatalities had attended his chase. But though similar disasters, however little bruited ashore, were by no means unusual in the fishery; yet, in most instances, such seemed the White Whale's infernal aforethought of ferocity, that every dismembering or death that he caused, was not wholly regarded as having been inflicted by an unintelligent agent.

### Change a different paragraph

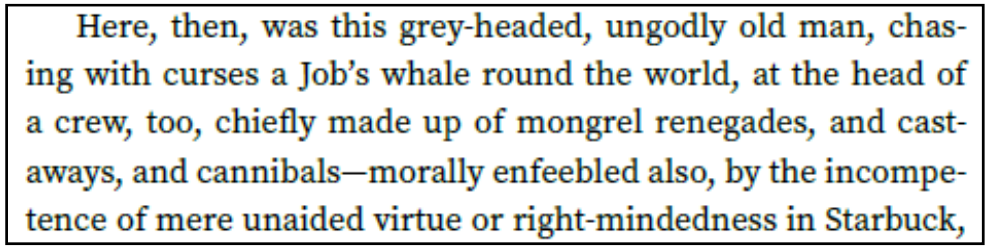
Here, the last line before a page break has 0.69em spaces. The first word on the next page is ‘incompetence’. The four lines before the page break cannot reasonably be squeezed enough to draw ‘incompetence’ back before the page break. A page should not end with a hyphen, so ‘incompetence’ also cannot be broken across the page. If there was one more line before the page break, then it would not be a problem if ‘incompetence’ was broken across the second-last and last lines of the page.


Here, then, was this grey-headed, ungodly old man, chasing with curses a Job's whale round the world, at the head of a crew, too, chiefly made up of mongrel renegades, and castaways, and cannibals—morally enfeebled also, by the

WHITENESS OF THE WHALE. 207

incompetence of mere unaided virtue or right-mindedness in Starbuck, the invulnerable jollity of indifference and reckless-

The chosen solution was to modify a previous paragraph so that it took up one less line and would leave room for one more line before the page break. There was a paragraph starting on page 204 that took 44 lines – or about 1¼ page – and ended with ‘able object.’ on its last line. When `letter-spacing.minimum="-0.006em"` is applied to that paragraph, it takes 43 lines. The rest of the lines on the page move up, and one extra line is pulled from the next page to fill that page. On the following page, one extra line of the problem paragraph is pulled from its second page. AH Formatter is not constrained by needing to avoid hyphenating ‘incompetence’ and so can format the paragraph differently:


Here, then, was this grey-headed, ungodly old man, chasing with curses a Job's whale round the world, at the head of a crew, too, chiefly made up of mongrel renegades, and castaways, and cannibals—morally enfeebled also, by the incompetence of mere unaided virtue or right-mindedness in Starbuck,

Losing a line like this can have a ripple effect on all of the following pages until something occurs to break the cycle. This could, for example, be a chapter starting on a new page or be a line that is unable to be pulled back a page because that would break the constraints of the ‘widows’ or ‘orphans’ properties. The change could also cause new analysis errors.

This change affected 78 pages, up until a page break where pulling back one line would leave a single widow line at the top of the next page. The number of lines pulled between pages varied considerably. As many as eight lines were pulled from one page to the previous. The number increased with paragraphs that had previously run short so that the minimum two lines had

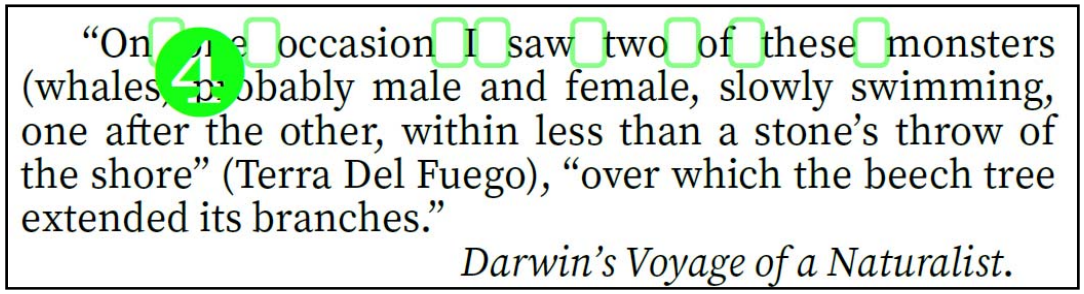
 

appeared after the paragraph broke across a page, and decreased with paragraphs where pulling more lines would cause too few lines on the following page. The number also increased where a chapter title was pulled to the preceding page. It decreased where the flourish at the end of a chapter was pulled to the preceding page and the next chapter's title moved to the top of the page it had been on: the flourish now had blank space below it, while the chapter title had more blank space above it than when it had immediately followed the flourish.

Despite 78 changed pages, there were no new errors reported, and eight fewer river errors.

### Both letter-spacing and word-spacing

This quotation has a line with 0.66em spaces. The solution to pulling '(whales)' back to the previous line was to specify both `letter-spacing.minimum="-0.01em"` and `word-spacing.minimum="-0.02em"`.




## Stage 7: River

Rivers are sometimes the easiest and sometimes the hardest errors to correct. A river totalling over 3em in a large paragraph was fixed by changing the minimum word-spacing by 0.2%. In other paragraphs, seemingly every 'word-spacing' and 'letter-spacing' adjustment just causes new rivers. This is particularly true for paragraphs with lines that contain many small words, because every change still has a good chance to line up the many spaces on the lines.

Most of the techniques for fixing rivers have been seen before: adjust one or both of 'word-spacing' and 'letter-spacing' or add a no-break space between words.

### River threshold

The river threshold for the entire document can be set by specifying 'axf:analyze-river' on the `fo:root` or by specifying 'river' in the Option Setting File. This example uses `river="2.5em"`, which detected the largest rivers without detecting more than it would be possible to fix in the time available.

### Keep multiple words together

Instead of adding a single U+00A0 NO-BREAK SPACE in a paragraph, it can be necessary either to use multiple U+00A0 to keep consecutive short words together:

```
<xsl:template
match="/TEI/text[1]/body[1]/div[1]/div[61]/p[6]/text()">
```

```
  <xsl:value-of select="ahf:no-break(., 'in the trough')" />
</xsl:template>
```

or to use multiple U+00A0 at different points in the same paragraph:

```
<xsl:template
match="/TEI/text[1]/body[1]/div[1]/div[3]/p[22]/text()">
  <xsl:value-of
      select="replace(replace(ahf:text(.),
                              'stood full', 'stood full'),
                      'a coffer-dam', 'a coffer-dam')" />
</xsl:template>
```

## Stage 8: Lines before and after

When chapters do not start on a new page, there can be a requirement for a minimum number of lines either before or after the chapter heading. Book Typography (6) recommends at least three lines above and below the chapter heading. This can usually be enforced using the widows and orphans properties, but not when, for example, the previous chapter ends with short lines of dialogue.

Specifying ‘axf:analyze-lines-before’ or ‘axf:analyze-lines-after’ on an fo:block indicates the required minimum number of lines before or after the formatted block. The Option Setting File does not include default settings for these because they are only useful on specific blocks.



## Set ‘widows’ and ‘orphans’

Setting ‘orphans’ on the first paragraph after the chapter title and setting ‘widows’ on the last paragraph in each chapter reduces the reported errors from nine errors to one.

```
<xsl:template
    match="p[preceding-sibling::*[1][self::head][@type =
'sub']]">
  <xsl:param name="atts" select="()" as="attribute()*" />

  <xsl:next-match>
    <xsl:with-param name="atts" as="attribute()*">
      <xsl:attribute name="orphans"
                     select="$analyze-lines-after" />
      <xsl:sequence select="$atts" />
    </xsl:with-param>
  </xsl:next-match>
</xsl:template>

<xsl:template
    match="div[@type = 'chapter']/p[last()]">
  <xsl:param name="atts" select="()" as="attribute()*" />

  <xsl:next-match>
    <xsl:with-param name="atts" as="attribute()*">
      <xsl:attribute name="widows"
                     select="$analyze-lines-before" />
      <xsl:sequence select="$atts" />
    </xsl:with-param>
  </xsl:next-match>
</xsl:template>
```

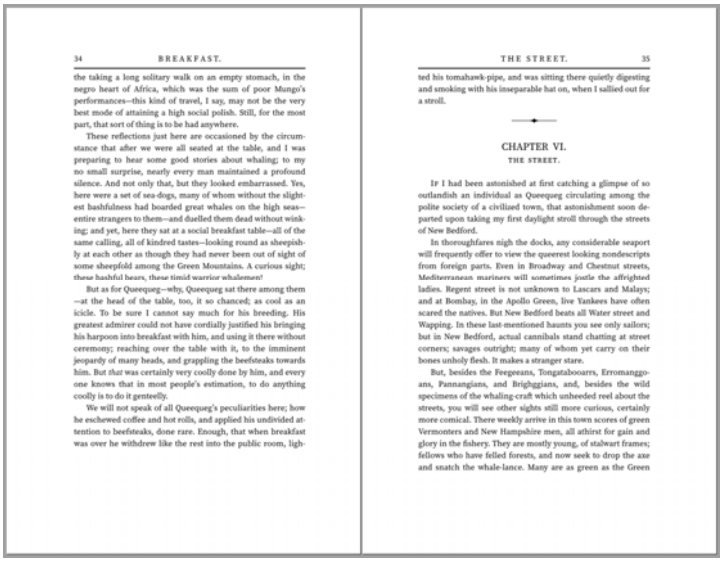
However, this affected 471 of the 635 pages of the body of the document, and the remaining lines error had not been an error before this change. This also caused a new river.

The ‘widows’ settings forced the correct number of lines:



It also created unbalanced spreads when lines were pulled from the preceding page:




## Force page break with <fo:block>

When there are not sufficient lines before a chapter title, one or more lines from the previous page can be 'turned over' onto the problem page. It is sometimes preferable turn over a line on a page two or more pages before the problem page just to avoid either creating unbalanced pages or having problems with orphans and widows.

An empty block, <fo:block/>, is a common technique for forcing a line break in formatted text. An <fo:block/> at the end of the second-last line of a page could force the last line onto the next page. However, when the block is justified, the text on the line with the <fo:block/> is treated like it is the last line of the block and is no longer justified. Specifying 'text-align-last="justify"' does not help: the last line of the portion after the <fo:block/> is also justified.

One solution is to enclose the portion on the second page in its own fo:block:

```
<xsl:template match="/TEI/text[1]/body[1]/div[1]/div[51]/p[3]">
  <xsl:param name="atts" select="()" as="attribute()*" />

  <fo:block text-align-last="justify" xsl:use-attribute-
sets="p">
    <xsl:copy-of select="$atts" />
    <xsl:variable name="text" select="ahf:text(text())" />
    <xsl:value-of
        select="substring-before($text, 'so many sails,')" />
    <xsl:text>so many sails,</xsl:text>
    <fo:block text-indent="0" text-align-last="left">
      <xsl:value-of
          select="substring-after($text, 'so many sails,')" />
    </fo:block>
  </fo:block>
</xsl:template>
```

## Stage 9: Unbalanced spreads

An unbalanced spread is two facing pages that each contains a different number of lines so that the text on one page ends higher up than the text on the other page. Opinions differ about the extent to which this is a problem. Requirements for Latin Text Layout and Pagination (LatinReq) (11) includes among its “The Classical Rules of Hyphenation and Pagination”:

*Balance facing pages by moving single lines.*

However, Book Typography (6) includes:

*The depth of the text panel is kept consistent throughout the book. However, in the US, some pages are set one line long or one line short as a way of manipulating the text and avoiding widows, orphans and bad word divisions.*

The First Edition has multiple unbalanced spreads.

## Conclusion

The automated analysis of formatting problems with AH Formatter V7.1 made it quick, simple, and straightforward to detect a range of problems in a 650-page book. The automated analysis is both quicker and more reliable than visual inspection of the formatted pages.

The errors, once found, were corrected programmatically by simple, repeatable transformations applied to the XSL-FO source of the formatted document and made without altering the original source XML.

Using the techniques described here, the number of reported errors has reduced from over 300 errors to just three errors,[2] with two of the errors also present in the First Edition:

| Error | Count | Pages |
|---|---|---|
| Blank pages at document end | 5 | 5 |
| Lines ending with same text | 1 | 1 |
| Paragraph widow | 2 | 2 |

The techniques include:

- Adding words to the hyphenation exception dictionary to affect all uses of those words.
- Adding no-break spaces before specific words in specific paragraphs.
- Adding zero-width joiner to inhibit hyphenation of specific words in specific paragraphs.
- Using 'hyphenation-push-character-count' to stop short hyphenated fragments.
- Using 'axf:hyphenate-hyphenated-word' to limit hyphenation to literal hyphens.
- Using 'hyphenate' to disable hyphenation.
- Adjusting word spacing to influence line breaking.
- Adjusting letter spacing to influence line breaking.
- Using extension properties, such an `axf:analyze-white-space="none"`, to ignore unavoidable errors.
- Changing an earlier paragraph to use fewer lines to change where the page break occurs in a paragraph that contains an error.
- Forcing a page break in a paragraph.

The general method for checking and correcting analysis errors is:

1. Finalize the stylesheet for the document: changes to the basic styles after any corrections have been applied risks making the corrections either irrelevant or the cause of new errors. If you find yourself thinking that you have invested too much into fixing errors to be able to make a text or style change that could undo some of the fixes, then you started the error fixes too early.
2. Run the analysis utility on the source XSL-FO or HTML document to get a baseline for the errors that are present.
3. Examine the analysis report to see the number and severity of the reported errors.
4. If necessary, adjust the error thresholds in the Option Setting File until the analysis utility report shows only the errors that are worth fixing with the time and resources available.
5. Start from the front of the document and work towards the back to fix (or ignore) the errors in sequence. It will be tempting to fix some of the biggest errors first, but those

[2] Ignoring the blank pages at the end of the document and after adjusting the error thresholds.

fixes could be undone, or could cause new errors, when other fixes are made. The example in this document of 78 pages changing because 1½ word was pulled back to the previous line shows how a fix can affect far beyond its intended target.

6. Reanalyze the document frequently to check that the changes have not caused new errors.

## References

1. **Melville Electronic Library.** Moby-Dick Side-by-Side: The American And British First Editions. *Melville Electronic Library.* [Online] https://melville.electroniclibrary.org/moby-dick-side-by-side.

2. **Indiana University.** Moby-Dick, or, The Whale (PDF). *Wright American Fiction.* [Online] http://purl.dlib.indiana.edu/iudl/wright/printable/VAC7237.

3. **Antenna House.** Automated Analysis. *AH Formatter V7.1.* [Online] https://www.antenna.co.jp/AHF/help/en/ahf-analyzer.html.

4. **The University of Chicago Press.** *The Chicago Manual of Style.* Chicago and London : The University of Chicago Press, 2017.

5. **Bringhurst, Robert.** *Elements of Typographic Style.* Vancouver, BC, Canada : Hartley & Marks, 2001.

6. **Mitchell, M. and Wrightman, S.** *Book Typography: A Designer's Manual.* Marlborough, Wiltshire : Libanus Press, 2005.

7. **Antenna House.** analysis-utility - AH Formatter Analysis Utility. *GitHub.* [Online] https://github.com/AntennaHouse/analysis-utility.

8. **Indiana University.** Moby-Dick, or, The Whale. *Wright American Fiction.* [Online] http://purl.dlib.indiana.edu/iudl/wright/encodedtext/VAC7237.

9. **Raptis Rare Books.** Moby Dick, or, The Whale Herman Melville First Edition Rare. *Raptis Rare Books.* [Online] [Cited: October 19, 2020.] http://web.archive.org/web/20201019122014/https://www.raptisrarebooks.com/product/moby-dick-or-the-whale-herman-melville-first-edition-rare/.

10. **MacKellar, Thomas.** *The American Printer.* Philadelphia : MacKellar, Smiths & Jordan, 1885.

11. **Antenna House.** Automated Analysis. *AH Formatter V7.0.* [Online] https://www.antenna.co.jp/AHF/help/v70e/ahf-analyzer.html.