The more complex signatures (especially sexto, octavo, and duodecimo) may require some practice to fold correctly. Remember that you will have to cut some of the creases to make the book open correctly. It's best to print out a signature or two separately and practice folding before printing the entire book.

10 Bugs

Multiple-signature sections in signatures other than folio and quarto do not work properly. They should.

There is at present no method to specify how many blank pages one would like at the front and how many at the back, because it's not clear to me why anyone would want to. But it would be good to have the option.

There is at present no method to deliberately make the book imperfect; for example, to round off a quarto book with a folio section on smaller paper. I'm not sure that there is any really good reason to do this anyway, but again, it would be good to have the option.

11 Get it!

https://gitlab.com/dgoodmaniii/makebook/

12 See also

pdfinfo pdflatex

Makebook
Impress pdf pages on sheets for folding and binding

Donald P. Goodman III[*]

June 2016

1 Synopsis

makebook [-**V**] [-**v**] [-**h**] [-**f**] [-**t** *signature type*] [-**n** *signatures per section*] [-**H** *height of target page*] [-**w** *width of target page*] [-**d** *horiz. delta*] [-**D** *vert. delta*] [-**m** *horiz. offset*] [-**M** *vert. offset*] [-**s** *scale*] [-**i** *input file*] [-**o** *output file*]

2 Description

Makebook A utility for taking a normal pdf and impressing multiple pages onto single pages for printing, folding, and binding as a booklet or book. Supports folio, quarto, sexto, octavo, and

[*] E-mail: dgoodmaniii@gmail.com

duodecimo signatures; scaling of source pages; explicit or implicit determination of target page dimensions; and more options. Non-destructive; the original file is not effected.

3 Requirements

- bash
- pdfLaTeX
- pdfinfo
- pdftk
- GNU coreutils (od, bc and dc)[1]

4 Definitions

page A single side of a single leaf. In a normal pdf document, each page counts as a single page; it probably also has a *folio*, or explicit page number, printed on it, though not necessarily.

recto A page printed on the front side of a leaf. Traditionally given an odd-numbered folio.

verso A page printed on the back side of a leaf. Traditionally given an even-numbered folio.

leaf What we commonly call the "page" of a book, which contains two pages. Different signature types yield different numbers of leaves; e.g., a folio produces two leaves with

[1] od, bc and dc can be installed through coreutils package: $ sudo apt-get install coreutils

It's easier to look at, easier to write, and produces better results. If you intend to print and bind yourself, *please* consider your final output size when you design your book; it'll make it easier for you and nicer for your readers.

It is often desirable to set a document which one plans to impose on a given size paper such that a larger size paper will be appropriate. The author, for example, had good luck imposing a 6in × 4.25in document into quarto signatures on legal size paper (8.5in × 14in) with the following:

```
makebook -v -i file.pdf -H 14in -w 8.5in -D 1in -t 4to
```

Initially, I had intended to set the book for quarto signatures on letter paper, but the fit was too tight; using legal paper instead allowed me to increase the page height from 5.5in to 6in and still have plenty of room for trimming and binding.

It is often helpful, when determining appropriate settings, to cut out a few signatures' worth of pages from the true document and to run the tests on that. **makebook** takes some time to run, and using fewer signatures can save a great deal of time.

Happy binding!

9 Cautions

When you're printing these documents on a duplex printer, keep in mind that the side on which you flip can have an effect on whether the pages line up correctly. For a quarto signature, for example, if your duplex printer gives you the option, tell it to flip on the "long" edge; but for an octavo, tell it to flip on the "short" edge.

To set a brochure of 16 pages or less for printing as a booklet. Run the following:

```
makebook -v -t octavo -i source.pdf
```

Both of these examples assume that your pages are properly sized to fit on your target sheet; e.g., that you have one-quarter letterpaper size pages to be set in quarto on a standard U.S. letterpaper sheet. If not, you may have to get more creative.

For imposing more-or-less standard LaTeX-produced pages in folio on U.S. letterpaper, the following has proved to be mostly serviceable:

```
makebook -i file.pdf -H 8.5in -w 11in -t 2o -v -d "-1.1in" -s 0.8333 -n 2
```

This provides minimal scaling of the page to maintain maximum quality, while still adjusting things as necessary to fit. It's likely that you'll need to do some tweaking, though.

For imposing pages originally designed for A4 paper in folio on A4, the following is at least worth trying:

```
makebook -v -i file.pdf -n 2 -t 2o -w 29.7cm -H 21cm -s 0.8
```

Again, it's quite likely that you'll need to do some tweaking of these measurements.

It's always *much* easier to impose pages in their original size, and it mostly looks a lot better, too. For example, the last example, when imposing A5 pages onto A4 paper in folio, looks like this:

```
makebook -v -i file.pdf -n2 -t 2o
```

four pages, while a quarto produces four leaves with eight pages.

signature Loosely used by makebook to mean a single sheet of paper, folded up to produce a given number of leaves and pages. Traditionally a signature was actually simply the mark put on a sheet to tell the binder in what order the sheets, once folded, should be bound, usually just an uppercase Roman letter or some similar mark; however, I've chosen to use "signature" to mean a single sheet, which may be equivalent to "section" (see below) or may not be. Sections may contain multiple signatures, but a signature is always unitary. Signatures can be of many different types, traditionally named by the number of leaves they produce. E.g., a folio signature produces two leaves; a quarto produces four; an octavo eight; and so on.

section A self-contained part of a book, properly folded and ready for binding. Sections may consist of one or more signatures, and a book may be made up of one or more sections.

delta Additional distance added between the impressed pages on the target sheet. Can be either vertical or horizontal. Useful for, e.g., adding the "binding correction" to the interior margins of the book. Can be positive or negative.

offset The extra distance added before placing the source page on the target page. Affects the placement of the top left corner of the source page. Can be vertical or horizontal, positive or negative.

5 Options

-V Print version and licensing information, then exit successfully.

-v Print verbose output while running. By default, **makebook** produces very little output, so that it can be included in a pipe; but if you're running **makebook** by itself, as is the usual situation, it can take a while to do its work, so this option is quite useful to keep you apprised of what **makebook** is doing.

-h Print a help screen, then exit.

-f Favor the front rather than the back when filling up sections with blanks; that is, put more blanks in the front than in the back. The default puts more blanks in the back than the front (unless the number is equal).

-t *signature type* What type of signature you want. You can use either the traditional term or the traditional abbreviation for the accepted types of signature:

folio, 2o folded once, for four pages per sheet.
quarto, 4to folded twice, for eigth pages per sheet.
sexto, 6to folded once, then cut, the new piece folded again; six pages per side of sheet.
octavo, 8vo folded three times, for sixteen pages per sheet.
duodecimo, 12mo folded twice times, then cut, the new piece folded again; twelve pages per side of sheet.

The user may give either the long or the short versions of these signatures types, separated above by commas, with no change in **makebook**'s effects. Defaults to "folio".

ever, it is necessary (at least in traditional binding) to have sections of equal numbers of leaves. A book so arranged is called "perfect"; a book not so arranged is called "imperfect."

To accomplish this, blank pages are added to the front and back of the book when necessary. (That is, to the first and last sections.) As far as possible, equal numbers of blank pages are added both to the front and back of the book; if uneven numbers of blank pages are required, one additional blank page is added to the back. If, however, that would result in an uneven number of blanks in the front, one additional blank page is added to the back and one fewer to the front. This ensures that odd-numbered folios will always correctly appear on recto pages.

One can, however, force **makebook** to favor the front rather than the back when unequal numbers of blanks are required to make the book perfect. This is specified with the "-f" option. With this option specified, **makebook** follows precisely the same procedure in determining the number of blanks; when it's done, however, it simply switches the number which goes in the front with that which goes in the back. Note that this sometimes results in the same number of blanks at the front and the back both with and without the "-f" flag set; for example, if the number of blanks is "2" at the front and "3" at the back. As the program is currently written, there is no way to avoid this.

8 Examples

To set a short eight-page document for quarto printing as a booklet. Run the following:

```
makebook -v -t quarto -i source.pdf
```

consequently gives PostScript points as dimensions when they are requested. Any dimensions given without units will therefore be assumed to be big points.

makebook will also accept other units, however; below are the units **makebook** understands. In parentheses after those units are the abbreviations which **makebook** knows; unless you specify the units with these abbreviations, **makebook** won't understand you.

inches	**(in)**
centimeters	**(cm)**
millimeters	**(mm)**
points	**(pt)**
picas	**(pc)**
big points	**(bp)**

Please note that "pt" is the traditional printers' point, not the PostScript "big point"; traditional printers' points are slightly smaller than big points, there being 72.27 of them per inch. Picas here is twelve (12) printers' points, not twelve big points.

To input these other units, simply attach their abbreviations directly to their numbers; e.g., "8.5in". **makebook** will recognize such measurements and convert them into bp for use internally.

7 Filling up sections

Sometimes, the number of pages in the source document doesn't match up with the number of pages necessary to set all the pages into sections each containing an equal number of pages. How-

-n ***signatures per section*** Provides the number of signatures of the given type which will be assembled into each section. Defaults to one. Note that this option only works for folio and quarto signatures; though **makebook** will not complain if you try it with others, your results will (probably) be wrong.

-H ***height of target page*** The height of the page onto which **makebook** will impress the pages of the original document. If omitted, **makebook** will simply multiply the dimensions of the original page according to the type of signature and use that, so as not to require any scaling of the source pages.

-w ***width of target page*** The width of the page onto which **makebook** will impress the pages of the original document. If omitted, **makebook** will simply multiply the dimensions of the original page according to the type of signature and use that, so as not to require any scaling of the source pages.

-d ***horiz. delta*** Alters the horizontal distance between impressed sheets on the target page. Defaults to 0.

-D ***vert. delta*** Alters the vertical distance between impressed sheets on the target page. Defaults to 0.

-m ***horiz. offset*** Requires **makebook** to place the upper left corner of each impressed page offset horizontally by this amount. Defaults to 0.

-M ***vert. offset*** Requires **makebook** to place the upper left corner of each impressed page offset vertically by this amount. Defaults to 0.

-s ***scale*** If you intend to scale the page, enter the scaling factor here. "1" will not scale at all; less than 1 will reduce the page's size, greater than 1 will increase it. Use this with care;

scaling nearly always degrades the quality of the result, no matter how careful you were with your vector graphics.

-i ***input file*** Specifies the name of the input file; that is, the file that **makebook** will be imposing onto a new document. If not provided, **makebook** will simply use standard input.

-o ***output file*** Specifies the name of the output file which **makebook** should use for the result of its operations. If omitted, **makebook** will prepend the string "sigs_" to the beginning of the input file name and use that. To use standard output, include this option, but give "stdout" for the filename.

6 Dimensions

6.1 Default dimensions

makebook, as will be seen below, permits the explicit statement of some dimensions. Absent such statement, however, default values will be assigned. Furthermore, the size of the input page cannot be set explicitly; it is always determined automatically by the script. This section explains a bit how these dimensions are determined.

THE SOURCE PAGE: The dimensions of the source page are determined very simply, by running "pdfinfo" through an awk filter and taking that at face value. This has worked well for the author's purposes. These dimensions cannot be set explicitly.

THE TARGET PAGE: The dimensions of the target page can be specified explicitly at the command line, with "-H" and "-w". If they are not, however, **makebook** does its best to give them sensible values. However, I did not want **makebook** to assume a given paper size (and thus begin a letterpaper vs. A4 firestorm) or to assume a desired scaling (and thus have a default which probably ruins page quality), so it does so very simply.

makebook simply multiplies the source page dimensions by appropriate values to ensure that the target page will be exactly the correct size to hold the requested type of signature assuming no scaling of the source page.

In other words, if a pdf source document is on halfletter paper (that is, 8.5in × 5.5in), and folio signatures are requested, the resulting document will be 8.5in × 11in (normal letter paper). If a pdf is on letter paper, and folio signatures are requested, the resulting document will be 11in × 17in (normal ledger paper). If a document is set on 5.5in × 4.25in paper (a quarter of a normal letter page), and quarto signatures are requested, the resulting document will be 8.5in × 11in (normal letter paper). The same applies, of course, to metric paper sizes, and even to abnormal page sizes. **makebook** was largely tested, for example, using a source document set on 4.25in × 6in paper, and by default produced an 8.5in × 12in signature sheet. (Better results were obtained by explicitly requesting 8.5in × 14in paper, or normal legal paper, with a vertical delta, or -D, of 1in.)

By default, both deltas (-d and -D) and offsets (-m and -M) are set at zero.

By default, scaling is also set at 0.

6.2 Explicit dimensions

makebook assumes that all dimensions are in "big points" (bp); that is, PostScript points, of which there are 72in an inch. This is easiest because pdf uses PostScript points internally, and pdfinfo