Export org-mode to docx with citations via pandoc

John Kitchin

# Export org-mode to docx with citations via pandoc

Pandoc continues to develop, and since [the last time](http://kitchingroup.cheme.cmu.edu/blog/2014/07/17/Pandoc-does-org-mode-now/) I wrote about it there is improved support for citations. We will use that to convert org documents to Word documents that actually have citations and a bibliography in them. This post explores using helm-bibtex to insert pandoc compatible citations, and then using pandoc to convert the org file to a word document (docx). We can define the format of citations that helm-bibtex inserts in a function, and tell helm-bibtex to use it when in org mode.

Here is that code. This is just to give me a convenient tool to insert citations with searching in my bibtex file. I think you could just as easily use reftex for this, or an ido-completing function on bibtex keys. See [Pandoc - Pandoc User’s Guide](http://johnmacfarlane.net/pandoc/README.html) for directions on citation format. The key is to format the cite links to the pandoc format.

(defun helm-bibtex-format-pandoc-citation (keys)  
 (concat "[" (mapconcat (lambda (key) (concat "@" key)) keys "; ") "]"))  
  
;; inform helm-bibtex how to format the citation in org-mode  
(setf (cdr (assoc 'org-mode helm-bibtex-format-citation-functions))  
 'helm-bibtex-format-pandoc-citation)

Now, we can cite the org-mode book (Dominik 2010), and some interesting papers on using org-mode (E. Schulte and Davison 2011; Eric Schulte et al. 2012). You could pretty easily add pre and post text manually to these, after selecting and inserting them.

We need a bibliography file for pandoc to work. I will use a bibtex file, since I already have it and am using helm-bibtex to select keys. I found pandoc could not read my massive bibtex file, perhaps it does not support all the types yet, so I made a special small bibtex file for this. So, now all we need to do is convert this file to a docx. I use a function like this to do that. It uses an org-ref function to get the bibliography defined in this file, derives some file names, and then runs pandoc.

(defun ox-export-to-docx-and-open ()  
 "Export the current org file as a docx via markdown."  
 (interactive)  
 (let\* ((bibfile (expand-file-name (car (org-ref-find-bibliography))))  
 ;; this is probably a full path  
 (current-file (buffer-file-name))  
 (basename (file-name-sans-extension current-file))  
 (docx-file (concat basename ".docx")))  
 (save-buffer)  
 (when (file-exists-p docx-file) (delete-file docx-file))  
 (shell-command (format  
 "pandoc -s -S --bibliography=%s %s -o %s"  
 bibfile current-file docx-file))  
 (org-open-file docx-file '(16))))

And now we run it to get our docx.

(ox-export-to-docx-and-open)

Here is the result: <file:org-to-docx-pandoc.docx>

It is not too bad. Not all the equations showed up below, and the figure did not appear for some reason. But, the citations went through fine. A downside of this is the citation links are not clickable (but see [Making pandoc links](id:2958EFDC-CC33-4E2A-8A92-D2BE06EBB3F2) for a way to do this), so they lack all the awesome features that org-ref gives them. Maybe pandoc can convert these to LaTeX links, but we already have such a good framework for that I do not see why you would want to do it. A better option is to figure out how to export the org file to an org file, and transform the org citation links to pandoc citations, then use pandoc on the temporarily transformed buffer. That way, you keep the cite links and their functionality, and ability to export to many formats, *and* get export to docx via pandoc.

There are other options in pandoc to fine tune the reference format (you need a csl file). That can be included in the org-file via file tags pretty easily. These citations are not links in the word document, and it does not look like they can be converted to footnotes, endnotes or interact with Endnote or Zotero at this time, but it is a step forward in getting a passable word document with references out of org-mode!

Since we are testing, let us try it some other typical features in an org-file.

## Numbered list

1. Item 1
2. Item 2
3. Item 3

## Bulleted list

* item 1
* item 2
* item 3
  + subitem

## definitions

org-mode

tool for awesomeness

## Math

One equation:

A second equation:

## An image

A little icon.

## A table

A little table.

|  |  |
| --- | --- |
| x | y |
| 1 | 2 |
| 3 | 4 |

a plain table

|  |  |
| --- | --- |
| x | y |
| 1 | 2 |
| 3 | 4 |

## Making pandoc links

Here I show a way to get clickable text on pandoc links. I found a nice library called [button-lock](https://github.com/rolandwalker/button-lock) that uses a regular expression to attach text properties to matching text.

Below I repeat the citations so it is easy to see the effect after running the code block. Indeed, you get clickable text, even org-ref like capability. I think you could even add the idle-timer messages, and the org-ref menu.

Now, we can cite the org-mode book (Dominik 2010), and some interesting papers on using org-mode (E. Schulte and Davison 2011; Eric Schulte et al. 2012). You could pretty easily add pre and post text manually to these, after selecting and inserting them.

You would need to make this code run in when you open an org-file to get it to work every time.

(require 'button-lock)  
(global-button-lock-mode)  
  
(button-lock-set-button  
 "@\\([-a-zA-Z0-9\_:]\*\\)"  
 (lambda ()  
 (interactive)  
 (re-search-backward "@")  
 (re-search-forward "@\\([-a-zA-Z0-9\_:]\*\\)")  
 (let\* ((key (match-string-no-properties 1))  
 (bibfile (cdr (org-ref-get-bibtex-key-and-file key))))  
 (if bibfile  
 (save-excursion  
 (with-temp-buffer  
 (insert-file-contents bibfile)  
 (bibtex-search-entry key)  
 (message (org-ref-bib-citation))))  
 (message "No entry found"))))  
 :face (list 'org-link))

## References

Dominik, Carsten. 2010. *The Org-Mode 7 Reference Manual: Organize Your Life with GNU Emacs*. UK: Network Theory.

Schulte, E., and D. Davison. 2011. “Active Documents with Org-Mode.” *Computing in Science Engineering* 13 (3): 66–73. doi:[10.1109/MCSE.2011.41](http://dx.doi.org/10.1109/MCSE.2011.41).

Schulte, Eric, Dan Davison, Thomas Dye, and Carsten Dominik. 2012. “A Multi-Language Computing Environment for Literate Programming and Reproducible Research.” *Journal of Statistical Software* 46 (3): 1–24. <http://www.jstatsoft.org/v46/i03>.