`lyxport` documentation

Mark Bravington, February 2025
updated for v1.0.182+

1 Overview

The `lyxport` R package is for exporting LyX to MSWord--- which I sometimes have to do, under duress--- and perhaps other formats. Unlike LyX's built-in "MS Word Open Office XML" export, Lyxport[1] does proper cross-referencing including tables, figures, lists, equations, appendices, and bibliography; and it sorts out a few other quirks too. Most of the heavy lifting is still done by Pandoc, as in LyX's built-in export option. However, Pandoc--- wonderful though it is!--- doesn't get everything right even with well-known filters (as you have probably discovered yourself by now, else you mightn't be reading this). So Lyxport contains a lot of *my* behind-the-scenes fiddly code in order to save *you* lots of manual post-fiddling.

The use-case I have in mind is basically my own: you have prepared a long and lovely paper in LyX that generates a perfect and pretty PDF. But for some reason you have to submit it to some journal that insists on miserable misbegotten manuscripts in MSWord. Sigh--- I sincerely feel your pain! There might be several rounds of back-and-forth between you and the journal, so you reeeally don't want to have to repeatedly do manual edits of the Word version of an 80% successful conversion using an editor you hate. The conversion should look reasonably decent, citations and cross-refs should all be correct, etc; but journals always fart around with the appearance of tables etc, so there's no need to get *every* detail of appearance exactly matching between PDF and MSWord. And that's the level that Lyxport aims at.

To see the features, do File->Open_example and open "examples/lyxport-demo.lyx". I haven't tested every LyX feature; it's mostly just stuff I need. More things might get added.

1.1 Setup

Once you've installed and loaded the `lyxport` package, read the helpery[2] in `?lyxprefhack`, make sure its requirements are met, and then try running `lyxprefhack()` to set things up for direct use from LyX. The helpery also explains how you can do the setup manually if `lyxprefhack` doesn't work— it hasn't been tested on all OSes, and people's computer setups are notoriously variable... The helpery also explains how to test whether it has all worked. If you are reading this document from inside LyX, then that's a good sign, but not a cast-iron guarantee— the real test is whether you can do "File->Export_as->MSWord_(lyxport)"? A good

[1] I don't approve of starting proper nouns with lower case. Causes more trouble than it's worth. Sure, software names might be spelt lower-case for reasonable reasons, but when referring to them, caps read more naturally in most cases. There's only one Pandoc.

[2] My word for "documentation"— as distinct from the word "document", which means I can use the abbreviation "docu" or "doco" exclusively for the latter.

test document is the example "lyxport-demo.lyx", which (if `lyxprefhack` or your manual setup has worked) you should be open in LyX via "File->Open_example" and searching. Try exporting it with LyX's built-in MSWord XML option, and then with "MSWord (lyxport)"; the differences will be obvious.

You can also see a PDF version of this document in R, via `RShowDoc("lyxport-docu",package="lyxport")`.

After setup, you won't normally use this package from R yourself; it will just be invoked from LyX. (The exception is if you want to use `requote_lyx` or (in future) any other "offline" support function. However, a keen user could experiment with the core function `lyxzip2word` (qv) directly, e.g. to produce other formats besides MSWord.

1.2 Do I need to modify my LyX document?

In terms of additions to your document: possibly not, unless you are using citations/bibliography. If you are, then you will need to add one line of ERT to define the bibliography style. Section 2) explains how, and mentions a few extra tweaks that you *might* want to do.

Also, you *might* want to add some ERT to improve the appearance of the exported file (section 5).

Apart from adding instructions to your document: there are general limitations on usable LyX features, some of which are stated below. So, you might need to change your document to not use features that don't work.

1.3 Are there are any other useful functions in `lyxport`?

The helper function `requote_lyx` tries to resolve inconsistent use of straight quotes, etc; you do have to use that manually from R, probably only once per LyX file. For example, I used it when creating this document, which was based on importing plain text files of R helpery. There's also those bibliography tweaks.

1.4 Versions

Version 1.0 of Lyport is written for Pandoc 3.5 and Pandoc-crossref v0.3.18.0.

It's quite possible that some of the problems which Lyxport 1.0 deals with, will actually be fixed in future releases of Pandoc. In that case, the Lyxport fixes might cause other problems..! Of course, I will try to maintain Lyxport accordingly, but I don't need Pandoc very often, so I might not get round to it quickly.

Note to myself really: what I should perhaps do for future-proofing, is make all the fix options switchable, and add a single character-string argument to `lyxzip2word` which controls all options (it would be disentangled inside `lyxzip2word`). Then users could easily-enough hand-modify the LyX Converter, to add an appropriate version of the argument in the call to `lyxzip2word`. But I haven't done this yet.

2 Bibliographies

Citations seem to work fine, ie whatever you specify in LyX for the PDF version also appears in the Word version— including the different options for parentheses, commas, etc (which took me a *lot* of hacking!). However, for the bibliography itself, Pandoc does not understand "styles" in the same way that Biber/Biblatex does (or

whatever animal of the Latex zoo it should be; I don't understand this stuff, and am happy to keep it that way). AFAICS none of the ways you can alter bibliography appearance in LyX/Latex will be understood. Instead, you'll need to specify a "CSL style" manually in your LyX source, as follows. First, you will need to get a suitable CSL file from the internet, eg via `https://editor.citationstyles.org/searchByExample` and store it somewhere (see below for thoughts about where). Second, you'll need some ERT to tell Lyxport (which will then tell Pandoc) about the CSL. Here's two options:

```
%% CSL journal-of-applied-genetics.csl
```

or

```
%% CSL ./journal-of-applied-genetics.csl
```

The first— i.e., with no path— is generally better; the CSL file is available "globally" for use by Lyxport on any LyX document. For that, you need to store the CSL file at the top of your biblio tree, ie in the folder "<top>" where "<top>/bibtex/bib/" contains the actual dot-bib file(s) you are using as biblio sources. (You would have told your Latex system about your biblio tree at some point. You might get lucky by trying `kpsewhich -var-value TEXMFHOME` in a shell. Or Google.)

The second option, with a "./", assumes you store the CSL file "locally", ie in the same folder as the LyX document that's mentioning it. In that case, you'll also need to insert the file as a Verbatim child document inside a Comment, like this:

The Comment prevents the contents of the CSL file from appearing in the rendered document (eg in PDF form), but the mention of a child-document tricks LyX into including the CSL file during File->Export->Lyx-zip (which is done automatically for you, as the first step in conversion to MSWord etc). (If I'd realized how Comments actually work, I would have used that instead of ERT for CSL-specification in the first place.)

If you specify several CSLs with that `%% CSL` mechanism, then only the final one will be used.

2.1 Capitalization

Biblatex/Biber seem to take account of capital letters without having to wrap them in {}, but MSWord and CSS don't, so that references which work fine with the former sometimes don't with the latter, unless tweaks are made. `lyxzip2word()` therefore contains code to forcibly capitalize references (i.e., changing the entries in a copy of the bibliography) where it looks like the user meant it. (Citations where the first word starts lower-case, e.g. for referring to a piece of Software, are also kept that way.) This might not be exactly the behaviour that *you* want, but *I* sure need it! So, you have to be careful to get your original biblio entries right; Capital in the database will mean Capital in the MSWord output, too.

2.2 Same author, multiple names

In a dot-bib file, the same author can appear with slightly different names in different papers: "A. Psmith", "Alan Psmith", "Alan B. Psmith", "A. Bertram Psmith", and so on. If you are not careful, your citations can

come out funny as a result. For example, you might see "Psmith et al. (1999)" but "A.B. Psmith (2004)" even though Alan Bertram is the only Psmith you are citing. With Biblatex and PDFs, you can suppress such nonsense via `uniquename=false` and `uniquelist=false`. But with CSL and MSWord etc, that (and pretty much everything :/) seems to be harder. `lyxzip2word()` sorts this out, rather forcibly, by calling `tidy_initials()` for you. The default behaviour is "lumping", so that "A.B. Psmith" and "A. Psmith" will be assumed to both be "A.B. Psmith" (but clear contradictions still won't be). You can turn that off by including one of these lines:

```
%% tidy_initials timid  # A.B.Psmith != A.Psmith
%% tidy_initials false  # leave my bib file alone!!!
%% tidy_initials gungho # the default
```

Under the hood: `lyxzip2word()` exports a mini-version of your dot-bib, containing only cited references. Then it runs `tidy_initials()` on that, in turn running `tex2utf8()` so that non-Latin characters hopefully get the same treatment as Latin ones. If you prefer to mod your bib at source beforehand, you can run either or both of those functions yourself. At your peril.

2.3 DIY bib-hacking

The CSL website I link to above is useful, but sometimes you just can't find a CSL that really does what you want; and editing an existing style seems incredibly difficult[3]. One *desperate* option might be to temporarily hack Lyxport's temporary copy of your own biblio file so that it Looks Nicer (in case you don't want to change your original bib file), just as `tidy_initials()` already does. For example, I wanted one particular institution to always be abbreviated, even though the bib file has it spelt out in full. So you can add your own R function to do that, along the lines below (it must be embedded in a Comment, as either ERT or perhaps a Program-Listing). The sole argument of the function should be a character vector (the text from the file), which your function should modify and return.

Comment

```
%% Bibhack¶
function( bibbo){¶
  insti <- grep( '^ *[Ii]nstitution *=', bibbo)¶
  IWC_full <- '.nternational +.haling +.ommission'¶
  iwc <- insti[ grep( IWC_full, bibbo[ insti])]¶
  bibbo[ iwc] <- gsub( IWC_full, 'IWC', bibbo[ iwc])¶
return( bibbo)¶
}¶
%% End Bibhack
```

[3] I did try. And I tried again. And I gave up. There are limits even to *my* obstinacy.

3 Other formats

My requirement is for MSWord, so that's what I've concentrated on. Things might or might not work in other formats; almost all the work in Lyxport is in generating a nice native-Pandoc document, and the final export step is just up to Pandoc, so there is a good chance other output formats will work at least reasonably. You can experiment inside R with the `outext` and `panoutopts` arguments of `lyxzip2word`. First, you will need to export your test document from LyX as a "LyX archive" (lyxzip).

FWIW I personally would prefer to use ODT rather than MSWord if I could (since ODT is open-source and the MSWord *program* can import ODT), but unfortunately LibreOffice's maths importing is broken (as of v7.6 and for some time before that). [One example: the vertical bar, which I use extensively for conditional probability. But there's other things too.] "Barring" that for-me-deal-breaking limitation, 'lyxzip2word' can export quite nicely to ODT, except figure sizes are not respected--- whether that's down to Pandoc's ODT writer, or to a limitation of ODT itself, I don't know. You'd need to manually resize all the figures within LO :/ Maybe there's an option in native Pandoc to specify figure sizes, in which case I could probably add code to do that--- but there's no point unless the ODT maths things get fixed.

4 How it works

In LyX, there needs to be a new "File Format" (in Preferences) which is an alias for the existing MSWord format. (The same would apply to any other desired type of output.) Then there needs to be a "Converter" from "Lyx-zip (archive)" format to the new alias. (These should be created automatically by `lyxprefhack`, or manually by you if necessary, as `?lyxprefhack` explains.)

When you pick "File->Export->[new format alias]", LyX will realize that the only way to produce the new format is to first export to Lyx-zip (producing, yes, a dot-zip or dot-tar-dot-gz file), then to run the new Converter. The latter launches R, and runs `lyxport::lyxzip2word` which does everything else, starting from that zip file. "Everything else" means: LyX -> Latex -> pre-process -> Pandoc-tex-to-native -> post-process -> Pandoc-native-to-MSWord-or-other.

This package is very effective despite *my* limitations: I don't understand Pandoc's internal document structure, I have zero wish to learn new programming languages or arcane file formats, and I have a pretty limited understanding of Latex--- much of the point of LyX being to avoid having to remember all the Latex gruesome details! However, the good news is that LyX exports a highly structured and limited form of Latex (unless you insert really nasty ERT...) which makes it easy to partially-parse the file using `grep` etc to find constructs that need attention. Similarly, Pandoc exports a highly-structured "native" format which IMO is much easier to partially-parse than JSON--- for example, indentation is highly consistent, so I can usually find the range of lines to `gsub` etc by matching indentation size. Then I monkeyed around (a lot) until things worked.

Because I rely on LyX's specific and tidy structuring of its Latex exports, `lyxzip2word` simply won't work on generic Latex documents.

5 Pandoc options

Pandoc uses an output-format-specific "template" or "reference-doc" to control some aspects of its output, eg fonts. See `https://pandoc.org/chunkedhtml-demo/6-templates.html` and `https://pandoc.org/MANUAL.html#optio` You might want to change the template; for example, Pandoc's default choices for MSWord "bold" looks pretty weird to me. The template/reference-doc also seems to be the place where you set features such as line- and page-numbering. I have included an example, "iwc-pandoc-reference.docx"[4], which started from the default that Pandoc describes in the above links; note that, to have any effect, you must edit the *style* of an item, not the appearance of the item itself. Then you must also tell Pandoc which template to use, by adding ERT like this (which again will not be visible in the PDF or DOCX version):

Some output formats, including MSWord, require a "reference-doc", while others want a "template"; so, use the appropriate word. As you see, you can specify different ones for different formats, which will be automatically selected depending on which Converter you invoke (if you have manually set up LyX Converters that use `lyxport` for more formats than MSWord). File location works the same way as CSS bibliography styles. No path means top of the TEXMF tree, "./" means same folder as the document, in which case you need to add something like this (again, invisible in PDF and DOCX):

Try to avoid filenames with spaces— here, and everywhere else!

6 Limitations

There are probably lots more than this. The list is in two parts, of which the first seems fairly permanent— it's structural stuff.

- Encoding must be set to UTF8 (Document->Settings->Language->Encoding).
- Labels must start with the right prefix for the thing they are labelling: "tab:" for table-floats, "fig:" for figure-floats, "eq:" for equations, "sec/subsec/par:" for sectioning, and "enu:" for lists. LyX will do this automatically for you when you set up a label, unless you perversely force it not to; so, don't do that.
  - Appendix title lines should have layout "section" (preferably; "part" *might* also work). Labels should therefore be "sec:<blah>" too.
- In order to be noticed, tables & figures have to be in floats with labelled captions. Each float can contain several actual tables or figures, but only one label. Requiring a label is reasonable; how else would you alert the reader to the existence and role of the table inside the main text?! Numbered sections and numbered equations don't need labels, unless they are cross-referenced.
- Figures: No absolute paths. (Otherwise, Lyx-zip-export uses lots of subfolders and I can't find the files.) I'm not sure about relative paths that go "upwards" either (eg "../sister/image.png").

[4] You should be able to find it from R with `system.file( 'examples/iwc-pandoc-reference.docx''');` or, from LyX, inside the `<userdir>/examples/lyxport` folder. Copy it somewhere else before actually using it.

  - `tikz` figures don't work (in the sense that Pandoc does not generate them on-the-fly when converting from dot-tex input file). You will need to generate them externally, then include them "manually" in your source somehow, like any "normal" graphics file. See eg `https://wiki.lyx.org/Tips/UsingTikZ-PGF`. [It might in future be possible to smooth this out a *bit*, e.g. via an ERT with special comments to load a pre-drawn graphic when `lyxport` is in use. Might.]
  - The same might well apply to other auto-drawing things, like Feynmann diagrams. Dunno.

- Equations inside tables won't be labelled.
- TOC & lists of Figures, etc. Since those are page-number-specific, they won't translate 100% between Latex & MSWord anyway.
- Multi-page figures using subfloat/ContinuedFloat *do* work, but you shouldn't put anything in the subfloat captions per se because they will not be printed (deliberately, Becoz Reasons). However, you can put things in the main caption of each continuation float.
- Your local "layouts" (LyX modules) won't be available unless you copy them somewhere else; see section 8. This is a LyX bug that might get fixed.
- Greyed-out text doesn't work (Pandoc doesn't support colored text). It's a shame, coz that leaves no way to preserve comments-as-comments in the dot-docx version.

Overall, the biggest limitation is Pandoc's Latex reader, but there are other problems too. For example: the "cases" environment gets spurious RH paren on export, LibreOffice/ODT maths has got problems, ... Anyway, for the second part of the list, here are some current limitations that I *might* fix in future:

- No Boxes (yet). In particular, resizebox and presumably its friends don't work (unfortunately, since I often use them to get tables to fit). Perhaps I should add code to delete them from the intermediate Latex files, so that at least something comes out.
- Table column widths are ignored.
- Only one Bibliography is produced.
- Equation numbering is either unsectioned (1, 2, 3 etc) or uses "section" as the prefix (1.1 etc). In principle it should also accept "chapter" or "part" for the prefix. Perhaps it should also allow user-specified arbitrary prefixes.
- Equations, Figures, and Tables in Appendices are numbered (A1,A2,...,B1,B2)--- no choice (and it's my preferred way). ¿Should I allow user-specced Apx encoding, eg S1, S2 instead of A, B? I don't know how to enforce any of this in LyX/Latex anyway.
- Lists always use LyX's default numbering hierarchy, eg 1-a-i-A. And there's no support for Enumerate-Resume yet.
- Citations are all author/year, rather than square-brackety. (Other than that, you can control citation appearance in the usual Latex ways.)

7 Requirements

This all needs Pandoc, ImageMagick (which at least on Windows should be installed automatically inside LyX's folder structure), and, of course, LyX itself.

The folders for the executables of Pandoc, LyX, and Rscript should be in the system path. If they aren't normally, then you can tell LyX to set them up automatically just within each LyX session, via "Tools->Preferences->Paths->PATH prefix".

I am going to assume that R_LIBS_USER and so on correctly set the libraries, ie the path(s) to R package folders.

8 The LyX userdir

There's a bug in LyX 2.4.3 (and some earlier versions) which means that, if you have a personal LyX userdir (which surely everyone does), it can't safely be made visible to LyX during the export operation. Normally this doesn't matter, because the userdir is mostly about interactive session stuff such as keybindings, but it does mean that any special modules in your "<userdir>/layouts" folder will not be visible. If you can't live without them, you will just have to copy them to your "<LyX-system-dir>/layouts" folder.

The LyX developers know about the bug, and apparently the very simple fix will appear in 2.4.4. Until then, for those who *really* want to know (be warned: do you *really*?), the problem is this: `lyxzip2word` needs to call a standalone LyX instance in order to export a Latex version of the source. (Most of the needed files--- the LyX source itself, any include-files, graphics--- are first exported into the Lyx-zip archive which is produced automatically, but the Latex version needs to be exported separately.) There is a handy CLI option "lyx --export latex <blah>" for that. Unfortunately, the latter can fall over if the userdir is set, either on the invocation line via "-userdir <blahblah>", or via the "LYX_USERDIR_24x" envar. (The problem is if there's a "uistyle" option within the "preferences" file— that's the bug that is being fixed.) Then LyX will crash with a SIGSEGV (and exit code 11 in Windows). When LyX calls Rscript, it *sets* that envar first--- which is normally good, coz it means that a converter could look at any userdir settings. However, when Rscript then calls LyX again, it will crash. So Rscript (ie `lyxzip2word`) has to *unset* that envar before calling LyX, which means that the second instance of LyX cannot know about your modules. You were warned!

9 FAQ

Q: *There was a bug, and then when I'd fixed it and tried again, it just seemed to be trying (and faililng) to convert the original file. What to do?*

A: This can occasionally happen with LyX's file cache not being cleared properly. It's a real pain. What you can try is going to the LyX tempdir, searching for your main dot-lyx and corresponding dot-zip file in all subfolders (they will probably be in "...tmpbuf<n>", where "<n>" is an unpredictable number), and deleting them. It mainly seems to happen to me when I'm debugging Lyxport and exporting dot-zips manually from LyX, so hopefully you won't encounter it...

Q: *I have exported my beautiful PDF to MSWord and now it looks like ****. What can I do about that?*

A: Well, yes, that's MSWord for you. But do look at "Pandoc options" above.

Q: *Why did you use R rather than writing a series of* Pandoc *filters in SDGL to translate FSDOIG format, like you're supposed to?*

A: Because I know R, and I have no interest in learning SDGL nor figuring out the intricacies of FSDOIG format. Also, modern R is really good for regexy manipulations--- people often don't realize that. If you don't already have R (which I expect many LyX users will), then it's not hard to install.

Q: *You clearly had to fix lots of issues. Why didn't you report them as bugs and wait for someone else to sort them out, like a logical systematic person would?*

A: Because I needed this for my own purposes, and fairly quickly. Other people are busy, and might not have the same view of what constitutes a "bug" or "something worth fixing" as I do. Like people say: "if you want something done properly, do it yourself". That said, I really ought to get round to reporting the bugs at some point--- but reporting bugs is quite time-consuming. I tried lots of Pandoc options involving `citeproc`, `eqnos`, `crossref`, and writen options, and just could not get things to behave, but I didn't record the failures in detail. If you try exporting "lyxport-demo.lyx" with LyX's built-in MSWord exporter, you'll get some idea...

Q: *Your approach based on counting indents etc is terribly fragile and Offensive To Logic. It is much better to build entire parse trees and then transmogrify the glorts using percortical DeMoivre contextual fermions.*

A: Errr, maybe so. But my approach *works*, and it works *now*, and *I* need it now!

Q: *Hmmm, I prefer to write all my documents in raw Latex because it is the Only True Way for ninth-level ninjas like myself; GUIs like LyX are for lesser mortals, monotremes, slime moulds, etc. However, I reluctantly concede that Lyxport does quite a good job of exporting to MSWord, and I note that it actually starts from a Latex file (as exported by LyX). Can I make use of it on my own superior Latex files?*

A: How very gracious of you. But to answer your question: no, not directly on general Latex files. There might be other tools Out There for that. Lyxport relies on the tightly-structured flavour of Latex exported by LyX. Your only chance would be to import your Latex into LyX, then export. It *might* all work, but expect pain.

Q: *Is this really a FAQ?*

A: Am I really Mickey Sodding Mouse?