

MultiMarkdown to LyX

C. R. Cowan

November 20, 2013

Part I

What is MultiMarkdown to LyX?

MultiMarkdown to LyX is an enhancement to MultiMarkdown that allows the production of a .lyx file. These files can be opened directly in LyX. As described on their web page:

LyX is a document processor that encourages an approach to writing based on the *structure* of your documents (WYSIWYM) and not simply their appearance (WYSIWYG).

LyX combines the power and flexibility of T_EX/L^AT_EX with the ease of use of a graphical interface. This results in world-class support for creation of mathematical content (via a fully integrated equation editor) and structured documents like academic articles, theses, and books. In addition, staples of scientific authoring such as reference list and index creation come standard. But you can also use LyX to create a letter or a novel or a theatre play or film script. A broad array of ready, well-designed document layouts are built in.¹

¹ <http://www.lyx.org>

MultiMarkdown already provides L^AT_EX support, so you may ask, why provide LyX output? There are two key areas where LyX stands out:

1. Editing and “Tweaking” the L^AT_EX product. While MultiMarkdown is powerful, it can’t support all of the capabilities of L^AT_EX. LyX provides a structured editing environment that will allow modifications to the final output without being a L^AT_EX expert. In addition it will make it easier to adapt to other document classes, beyond Memoir and Beamer. For example, I really like this “Tufte-Handout” class.²
2. Easy choice of output formats. The `view` and `export` functions of LyX provide a large number of output choices to include PDF, postscript, DVI, different HTML formats, and rich text. For example, the output to HTML using eLyXer is very attractive.

² Inspired by Edward Tufte, author of four beautiful books on design: <http://www.edwardtufte.com/tufte/>
More information on the tufte-handout class here: <http://wiki.lyx.org/Layouts/Tufte-handout>

Please note that the vast majority of my work has been using the LyX PDF (LuaT_EX) output processor and have been successful. Given the packages utilized, with other output processors “your mileage may vary.”

Part II

How do I use MultiMarkdown to \LaTeX ?

The \LaTeX support is built into MultiMarkdown. You specify \LaTeX output using the command line option `-t` or `-to` and specifying “ \LaTeX ”:

```
multimarkdown -t lyx
```

In general, the metadata in the MultiMarkdown file will control the details of creating the `.lyx` file.

1 \LaTeX Packages

Depending on the needs of the document class you are using and some of the metadata options, particular classes will be required.

MultiMarkdown to \LaTeX assumes the following packages and will add them to the `latex preamble` of the \LaTeX document settings dialog:

- `listings` – used in formatting codeblocks
- `natbib`
- `nomenc`
- `booktabs`
- `beamer-fragile` (only if you are using Beamer)
- `varioref` – to improve page references in links

Be sure that these packages are installed in your \LaTeX configuration before you attempt to produce output from \LaTeX .

In addition to the packages automatically included by MultiMarkdown, you can request that additional packages be added to this list via the `Packages` metadata entry (see *Packages* (Subsection 2.14 on page 7)).

Part III

What is different about \LaTeX documents?

When you are creating a document that will be processed by \LaTeX there are some differences from normal MultiMarkdown processing – and even \LaTeX targeted MultiMarkdown.

Most of the differences are managed by utilizing the metadata at the beginning of the MultiMarkdown document. These options are described in *MetaData* (Section 2 on the current page).

Because of the way LyX utilizes BibTeX, and to remain consistent with BibTeX formatting, the formatting of citations is a bit constrained.

2 *MetaData*

The LyX processor only looks at particular metadata entries. Other entries not described below are ignored, and you can leave them in place if you will be switching to other processors.

It also re-uses some of the L^AT_EX metadata entries to the same purpose as the L^AT_EX processor with a similar effect, but perhaps without all of the options.

2.1 *Abstract*

The **Abstract** metadata encloses the value provided in the LyX **Abstract** environment. It will be placed in the document as defined by the document class.

2.2 *Affiliation*

The **Affiliation** metadata encloses the value provided in the LyX **Institute** environment. It will be placed in the document as defined by the document class. This metadata entry is *only* processed for the **beamer** class.

2.3 *Author*

The **Author** metadata encloses the value provided in the LyX **Author** environment. It will be placed in the document as defined by the document class.

2.4 *Base Header Level*

The **Base Header Level** metadata element for LyX provides the same function as that element for L^AT_EX. The value defines the level within the LyX class that maps to the highest level (one hash mark) in the MultiMarkdown.

This metadata element can be used in conjunction with the Lyx **Headings** element.

2.5 *Bibtex*

The **BibTeX** metadata entry specifies the name of the BibTeX database to use for citations.

2.6 Class Options

`Class Options` is a convince feature. It consists of a comma separated list of options for the particular class chosen using the `Latex Mode` metadata entry.

This entry allows the specification in the MultiMarkdown metadata so that it does not have to be specified in LyX each time the MultiMarkdown file is processed.

2.7 Clean PDF

`Clean PDF` is a convenience feature that adds `hidelinks` to the class options. This removes the rather ugly magenta and cyan boxes around links in PDF files. The links still function but the outline is gone.

Instead of using this MetaData entry, you could add `hidelinks` to the `Class Options` MetaData entry if you prefer.

This entry allows the specification in the MultiMarkdown metadata so that it does not have to be specified in LyX each time the MultiMarkdown file is processed.

2.8 Date

The `Date` metadata encloses the value provided in the LyX `Date` environment. It will be placed in the document as defined by the document class.

2.9 L^AT_EX Input

While the `Latex Input` metadata entry for LyX looks similar to the entry for L^AT_EX, it really only has one function for LyX. Since `natbib` is used by default in LyX, either citations need to be compatible with ‘Author-Year’ (see *Citations* (Section 3 on page 8)) or you can specify the value `mmd-natbib-plain` for `Latex Input`. This changes the citation style in `natbib` to numeric to avoid errors. This is the same as is described in the MultiMarkdown User’s Guide³.

Note that use of this metadata entry with LyX *does not* include the contents of the `mmd-natbib-plain` file in the LyX document. It simply changes the citation style.

2.10 Latex Mode

The LyX `Latex Mode` metadata entry is very similar to the L^AT_EX entry. In both cases it selects the type of formatting to be performed and both default to `memoir`. The key difference is that the LyX version is really a LyX class. These tend to have the same name as their L^AT_EX counterpart.

In this case the value supplied *must* be the file name of a `.layout` file in the LyX `resources\layouts` directory. That name may not match the text in the `document->settings` dialog. You may need to browse that directory

³ Fletcher T. Penny
<http://fletcherpenny.net/mmd>

to find the proper filename. Just use the `name` portion of the filename and don't include the `.layout` extension.

For example the correct value for “memoir” is `memoir`, for “beamer” is `beamer` and for “handout (tufte)” is `tufte-handout`.

Because of the flexibility of using the `Lyx Headings` metadata entry, you should be able to accommodate many of the available LyX classes. Of course, there may be features of those classes you don't have access to through MultiMarkdown, but once you have your “draft” from MultiMarkdown loaded into LyX you can make further editing changes and take advantage of the full capabilities of the class.

Using `beamer` as a value for `Latex Mode` is a special case. This, indeed, will select the LyX `beamer` class. But, because `beamer` is a rather unique \LaTeX /LyX class, it utilizes special processing during the MultiMarkdown LyX conversion. Therefore, for the `beamer` case, the `Lyx Headings` metadata entry is ignored.

2.11 Lyx Headings

When MultiMarkdown is used to create \LaTeX output, users have a metadata entry called `Base Header Level` which controls which MultiMarkdown heading level corresponds to a \LaTeX heading level. The `Base Header Level` defines how far down the \LaTeX hierarchy the first MultiMarkdown level corresponds.

Because the LyX support is intended to encompass a broader set of LyX classes than just `Memoir` and `Beamer`, a metadata entry called `Lyx Headings` is provided. The value is a comma separated list of LyX environments in order from the highest level to the lowest level. Up to seven levels can be provided. If fewer than seven items are provided, they replace the equivalent number with the rest remaining the defaults.

The default set are:

1. Part
2. Chapter
3. Section
4. Subsection
5. Subsubsection
6. Paragraph
7. Subparagraph

While this is fine for `Memoir` and many other classes, it is not appropriate for others. For example, to use the `tufte-handout` class I supplied the metadata entry:

Lyx Headings: Part, Section, Subsection

This results in a LyX environment heirarchy of:

1. Part
2. Section
3. Subsection
4. Subsection
5. Subsubsection
6. Paragraph
7. Subparagraph

Chapter is thus missing and matches the `tufte-handout` class. Since I am only using three levels it is sufficient to leave the other entries as they are.

Please note that these items are case sensitive and must match the way that the environments appear in the environment selection dropdown. Do not append an asterisk to the environment name unless you provide the metadata `Number Headings` value of `yes`. Otherwise two asterisks will appear. I recommend that you do not provide asterisks in this list and use the `Number Headings` to control numbering.

Keep in mind that you can also use `Base Header Level` as well to control the mapping of the first level in the MultiMarkdown to a lower level in the LyX environment hierarchy.

2.12 Modules

The `Modules` metadata entry is a convenience feature. The value for this metadata is a comma separated list of `.module` file names. The value supplied *must* be the file name of a `.module` file in the LyX `resources\layouts` directory. Each entry will appear in the LyX document settings `Modules` section. **Note** that this is the file name of the module file *not* the text that appears in the LyX document settings `Modules` section. For example if you want the module described in the dialog as `Fix LATEX` you must specify `fixltx2e`.

This entry allows the specification in the MultiMarkdown metadata so that it does not have to be specified in LyX each time the MultiMarkdown file is processed.

2.13 Number Headings

LyX users can choose to have headings automatically numbered, or they can remain unnumbered. In LyX this is done by selecting an “environment” with or without a trailing asterisk. Environments without an asterisk are numbered, those with an asterisk are not numbered.

Thus choosing the `Section` environment results in a numbered header and choosing the `Section*` environment results in an unnumbered heading.

In MultiMarkdown this is controlled with the `Number Headings` metadata entry. If a value of `no` is provided, headings are not numbered. Any other value (or not providing this metadata entry) results in numbering the headers.

If numbering is requested, links will reference the numbering associated with the target of the reference. If numbering is turned off, appropriate links to the text of the target will be used.

Please note that if you turn off numbering, and use LyX to create a PDF file, it will not have table of contents bookmarks, making your PDF more difficult to navigate.

2.14 Packages

The `Packages` metadata entry is a convenience feature. The value for this metadata is a comma separated list of package names. Each name will appear in the LyX document settings `latex preamble` section in the format `\usepackage{xxx}` where `xxx` is the name of the package specified.

This entry allows the specification in the MultiMarkdown metadata so that it does not have to be specified in LyX each time the MultiMarkdown file is processed.

2.15 Quotes Language

This is used to specify which style of “smart” quotes to use in the output document. The available options are:

- `dutch`
- `english`
- `french`
- `german`
- `germanguillemets`
- `swedish`

The default is `english` if not specified.

2.16 Sub Title

The `Sub title` metadata encloses the value provided in the LyX `Subtitle` environment. It will be placed in the document as defined by the document class. This metadata entry is *only* processed for the `beamer` class.

2.17 Theme

The Theme metadata entry for \LaTeX has exactly the same function as for \LaTeX – it specifies the Beamer theme to be used for the creation of Beamer slides. This metadata entry is only used if `beamer` is the metadata value for `Latex Mode`.

3 Citations

Because \LaTeX uses `natbib`, in order to be compatible citations must be in a particular format known as the ‘Author–Year’ format. This format requires the Author’s last name followed by the year in parenthesis.

The example in the MultiMarkdown User’s guide:

```
This is a statement that could be attributed to its source [p. 23][#Doe:2006]
```

And following is the description of the reference to be used in the bibliography.

```
[#Doe:2006]: John Doe. *Some Big Fancy Book*. Vanity Press, 2006.
```

Would be re-written as:

```
This is a statement that could be attributed to its source [p. 23][#Doe (2006)]
```

And following is the description of the reference to be used in the bibliography.

```
[#Doe (2006)]: John Doe. *Some Big Fancy Book*. Vanity Press, 2006.
```

Resulting in:

This is a statement that could be attributed to its source Doe (2006, p. 23)

Alternatively you could use the `Latex Input` metadata element described in *\LaTeX Input* (Subsection 2.9 on page 4) to change the citation style to numbers.

4 Tables

The standard \LaTeX footnote command `\footnote` doesn’t work in normal tables⁴. Therefore, when MultiMarkdown converts a document to \LaTeX format, all tables that are created are \LaTeX “long tables.” Not only do long tables support proper footnotes, they also can span pages if required. \LaTeX provides a number of options for controlling long tables and you can take advantages of these once in the \LaTeX environment. MultiMarkdown sets the expected default options.

⁴ For \LaTeX see <http://wiki.lyx.org/LyX/Tables#footintab>

For \LaTeX see <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=footintab>

5 Captions

Unfortunately, \LaTeX doesn’t support footnotes in captions. MultiMarkdown to \LaTeX will place the footnote in the \LaTeX file in the event that \LaTeX

elects to support footnotes in captions but they will not appear in the current implementation.

6 Links and Labels

LyX MultiMarkdown generates a LyX label for every heading, image, etc. In LyX these labels look a bit ungainly, mostly looking like all of the text lowercased and strung together. Don't worry about that. If you edit in LyX you can still find the label when you insert a cross-reference. LyX MultiMarkdown properly proceeds all of the labels generated with a prefix that describes the type of label. For example `fig:` for a figure or `secti:` for a section.

If you stick with cross-referencing in MultiMarkdown, you have the advantage of being able to use a nice format for the reference. For example, should you use this cross-reference in this document:

You will find more information about this tool in [*What is MultiMarkdown to LyX?*]

It will appear like this:

You will find more information about this tool in *What is MultiMarkdown to LyX?* (Part I on page 1).

Note the italics.

If you use the `Number Headings` metadata entry to turn off numbering, LyX MultiMarkdown will place a reference to the name of the labeled item in quotes and linked to the labeled item.

Note that the `varioref` package is used to create page references and `\newref` statements are provided in the `LaTeX` header section of the LyX document -> `settings` dialog. You can modify these if required.

7 Code Blocks and Fenced Code Blocks

Code blocks and Fenced Code Blocks are implemented using the `LaTeX` “Listings” package. Basic settings for listing are used and you can “tweak” those (such as including line numbers) by opening the `settings` of the inset in LyX and making changes in the dialog box.

If you choose to use Fenced Code Blocks, you can specify a language to provide syntax highlighting. The language specified (following the three ‘tic’ marks) is case sensitive and must match the way the language is shown in the drop down box on the `settings` for the inset. For example `perl` must be specified as `Perl`.

Here is a C example

```
/* print_lyx_string - print string, escaping and formatting for LYX */
void print_lyx_string(GString *out, char *str, scratch_pad *scratch, short environment) {
    char *tmp;
    if (str == NULL)
        return;
    if (environment == LYX_PLAIN) {
```

```

    g_string_append(out, "\n\\begin_layout_Plain_Layout\n\n");
}
while (*str != '\0') {
    switch (*str) {
        case '\\':
            g_string_append(out, "\n\\backslash\n\n");
            break;
        case '\"':
            if (environment == LYX_PLAIN){
                g_string_append(out, "");
            } else {
                g_string_append(out, "\n\\begin_inset_Quotes_erd\n\\end_inset\n");
            }
            break;
        case '\n':
            if (environment == LYX_PLAIN) {
                if (*(str+1) == '\0'){ // skip last new line
                    break;
                }
                g_string_append(out, "\n\\end_layout\n\n\\begin_layout_Plain_Layout\n\n");
            } else {
                tmp = str;
                tmp--;
                if (*tmp == '_') {
                    g_string_append(out, "\n");
                } else {
                    g_string_append(out, "\n_"); // add a space
                }
            }
            break;
        case '<': // look for HTML comment LaTeX escape
            if ( (environment != LYX_CODE) && (environment != LYX_PLAIN) && (strncmp(str, "<!--", 4) == 0)){
                str+=4; // move past delimiter
                g_string_append(out, "\n\\begin_inset_ERT\nstatus_open\n\n\\begin_layout_Plain_Layout\n\n");
                while(strncmp(str, "-->", 3) != 0){
                    switch (*str){
                        case '\\':
                            g_string_append(out, "\n\\backslash\n\n");
                            break;
                        case '\"':
                            g_string_append(out, "\n\\begin_inset_Quotes_erd\n\\end_inset\n");
                            break;
                        default:
                            g_string_append_c(out, *str);
                    }
                    str++;
                }
                str+=2; // and past the end delimiter
                g_string_append(out, "\n\n\\end_layout\n\\end_inset\n");
            }
            else {
                g_string_append_c(out, *str);
            }
            break;
        default:
            g_string_append_c(out, *str);
    }
    str++;
}
if (environment == LYX_PLAIN) {
    g_string_append(out, "\n\\end_layout\n");
}

```

```

}
}

```

8 Raw L^AT_EX

Raw L^AT_EX to be included in your LyX document is specified using HTML comments. This is the only HTML that is processed. All other HTML is ignored.

The L^AT_EX provided in the HTML comment is presented to LyX as *Evil Red Text*, or ERT.

For example, if you input: `C<!--\sharp$->` or you will `B<!--\flat$->`

It will appear as: C# or you will B♭

Beware where you enter Raw L^AT_EX. If it is in an inconsistent place within your LyX document, LyX may fail to read the file correctly. If it is incorrect L^AT_EX, or if it isn't supported within the classes you are using, LyX may fail to produce a document.

Part IV

Tips and Tricks

9 Beware of HTML

Use caution when including any HTML in a document that will be converted to a LyX document. Regular HTML will be ignored so it can be included – but it won't appear in the LyX file or in LyX output. If, though, HTML comments are part of the HTML included, these will be interpreted as L^AT_EX in which case it is very likely that LyX will not be able to read the file, or that when LyX process the file it will fail to produce any output.

10 You can use many classes

LyX supports a large number of document classes and generally adjusts to handle those classes. Therefore it is likely that you can use a class that interests you in creating LyX from MultiMarkdown.

To use a class other than Memoir, simply include as a value in the `Latex Mode` metadata entry. Be sure to follow any of the caveats described in *Latex Mode* (Subsection 2.10 on page 4).

To be successful you will have to understand the limits of the class. This may require an adjustment to the class hierarchy. You can do this through the use of the `Lyx Headings` and `Base Header Level` metadata entries.

11 Beware of document class limitations

Having said that you can use a number of document classes, you may run into problems with their implementation. You will have to review any \LaTeX error messages if your document does not process correctly.

For example, in creating this document, using the `tufte-handout` class, I wanted to use the title “MultiMarkdown to \LaTeX ”. \LaTeX converts the letters `L-y-X` to the \LaTeX logo in a format that the `title` environment of this class could not handle and therefore I received \LaTeX errors when I tried to print the document.

My solution was to examine the \LaTeX produced to see how the logo is created. I then utilized the raw \LaTeX capabilities to create the title. Here is the metadata I used:

```
Title: MultiMarkdown to <!-- L\kern-.1667em\lower.25em\hbox{Y}\kern-.125emX\ -->
```

This resulted in the title you see in the printed document (and the Evil Red Text in \LaTeX).

12 Take advantage of \LaTeX

Once your document is in \LaTeX , you can take advantage of all of the editing functions available to tune your document. For example should you have decided to use one of the Tufte layouts you might like to use fonts similar to those used in the book. First download and install the Bergamo font⁵.

When you have converted your document to \LaTeX , in the Documents -> Settings -> \LaTeX Preamble paste the following:

```
\usepackage{fontspec}
\setmainfont[Mapping=tex-text,Numbers=OldStyle]{Bergamo Std}
\setsansfont[Mapping=tex-text,Numbers=OldStyle,Scale=MatchLowercase]{Gill Sans MT}
\setmonofont[Mapping=tex-text,Scale=MatchLowercase]{DejaVu Sans Mono}
```

In addition to changing the fonts, I also overrode the Tufte classes that suppress numbering (to allow more effective cross-referencing) by including the line:

```
\setcounter{secnumdepth}{3}
```

References

John Doe. *Some Big Fancy Book*. Vanity Press, 2006.

⁵ <http://www.fontsite.com/2077/bergamo-std/>