

Using the `ndsu-thesis-2022` L^AT_EX class — Documentation

Aaron Feickert^a, Jonathan Totushek^a, and C. Igathinathane^{b,*}

* Maintainer, Bug Reports and Enquires: Igathinathane Cannayen (i.cannayen@ndsu.edu)

Github: <https://github.com/CIgathi/NDSU-Thesis-Class.git>

^aDepartment of Mathematics, NDSU

^bDepartment of Agricultural and Biosystems Engineering, NDSU

5 December 2022

Contents

1	Introduction	2	11	Equations	13
2	Using and installing L^AT_EX — online and desktop environments	2	12	References/Bibliography	14
3	Thesis Example	2	12.1	Using BibTeX	14
4	Documentclass Options	6	12.1.1	Cite while you write (CWYW) using natbib	14
4.1	Disquisition degree and type	7	12.1.2	Bibliography generation and files handling	14
4.2	Font size	7	12.2	Using BibL ^A T _E X	15
4.3	Auto-numbered, chapter-numbered, and unnumbered styles	7	12.2.1	Commands, cite, bibliography generation and files handling	15
4.4	Paragraph text justification	7	12.2.2	Individual chapters bibliography (multiple)	15
4.5	Draft and display document frames	7	12.3	Specialization specific bibliography — ASABE and IEEE examples	16
4.6	Fonts	8	13	Appendix	17
4.7	Individual or combine reference listing	8	13.1	Single and multiple named appendices	17
5	Preamble Information	8	13.2	Appendix figures and tables	17
5.1	General information - packages and shortcuts	8	13.3	Closing appendices and creating TOC, LOAT, and LOAF	17
5.2	Dissertation front pages	9	14	Additional Information I — Special Commands	18
5.2.1	Title	9	14.1	Whole document text spacing	18
5.2.2	Author	9	14.2	Tables: Advanced commands	18
5.2.3	Major department/program choice	9	14.2.1	Table row spacing and fonts	18
5.2.4	Department or program	9	14.2.2	Landscape tables	18
5.2.5	Degree option	9	14.2.3	Long tables	18
5.2.6	Date	9	14.2.4	Landscape long tables	18
5.2.7	Examining committee	9	14.3	Figures: Advanced commands	19
5.2.8	Approval information	9	14.3.1	Figures in separate folder	19
5.3	Dissertation front matter	10	14.3.2	Flowchart - tikz package	19
5.3.1	Abstract	10	14.3.3	Subfigures	19
5.3.2	Acknowledgements	10	14.3.4	Landscape subfigures	19
5.3.3	Dedication	10	14.3.5	Multiple page subfigures	19
5.3.4	Preface	10	14.4	Chemical symbols	20
6	Automatic Components	10	14.5	Annotation commands	20
6.1	Table of contents	10	14.6	Clever reference — cross-referencing items and labels	20
6.2	List of tables and figures	10	14.7	Bibliography compilation issues	21
6.3	List of abbreviations	10	14.8	Proper development of bibliography bib file	21
6.4	List of symbols	11	14.9	Temporary ending	22
6.5	List of appendix tables and figures	11	14.10	Spacing adjustment around non-textual elements	23
7	Beginning the Document	11	14.11	Line numbers for the whole document	23
8	Headings	11	14.12	Chapter styles	23
9	Dummy Text and Images	12	14.13	Chapters as individual files	24
10	Tables and Figures	12	14.14	Chapters as subfiles	24
10.1	Tables	12	14.15	Defining and using specific commands, environments, and packages	25
10.1.1	Full-width tables — <code>tblr</code> environment from <code>tabularray</code> package	12	14.16	Thesis to journal article conversion for submission	25
10.1.2	Full-width tables — manual method	12	14.17	Advanced options	26
10.2	Figures	13	14.18	Limitations of the class	26
10.2.1	Shortcuts for figures — direct and optional	13	15	Additional Information II — Some Tips For Customization	27
10.3	Captions	13	15.1	General suggestions	27
			15.2	Voice of the T _E X and L ^A T _E X developers!	27
			Acknowledgements	28	
			Index	28	

1 Introduction

The `ndsu-thesis-2022` L^AT_EX is an updated version of the previous `ndsu-thesis` class file. This class generates disquisitions intended to comply with the disquisition requirements of the North Dakota State University (NDSU) Graduate School. This class is not officially endorsed by NDSU or the NDSU Graduate School, but efforts are underway toward that goal. It should be noted that several theses and dissertations were made and got approved by the Graduate School using the NDSU L^AT_EX thesis class in the past. Since disquisition requirements are subject to change at any time, the user is advised that the most current disquisition style policies supersede this class. However, following the Graduate School approved template and collected experience from several previously approved dissertations, this L^AT_EX class was coded to incorporate the various required features and lessons learned. To ensure compliance with all NDSU Graduate School requirements, the user is encouraged to consult the NDSU Graduate School webpage and the links provided for detailed requirements and guidance on disquisition formatting guidelines, templates, section formatting, and examples (https://www.ndsu.edu/gradschool/current_students/graduation/theses_dissertations_papers/disquisition_formatting).

The bundled template or the thesis example given (Section 3) can be used as an easy starting point for using the class. Modification of the class file’s code may result in unexpected behavior and is at the user’s own risk. We recommend including additional packages and commands in the source file (*.tex) itself for the desired customization as required by the departments and the users.

2 Using and installing L^AT_EX — online and desktop environments

Several online (e.g., Overleaf, Kile LaTeX Editor, Authorea, Papeeria, and so on) and standalone desktop versions (e.g., TeXMaker, TeXWorks, TexShop, TeXStudio, and so on) of L^AT_EX editors are available. Online editors are “ready-to-go,” with several templates, tutorials, and help documentation, where the user need not install the software but require an internet connection. The desktop version requires software installation and updating (not very frequently). Resources (text and video instructions) are available on both how to use the online editor and install the L^AT_EX desktop version of users’ choice. As L^AT_EX is open source, most of these editors are free.

3 Thesis Example

Below is a brief example of an M.S. thesis (copy/load code in editor, have necessary resources [class + figures], and compile for output) that includes all required and several optional elements. An attempt was made to cover most of the aspects (prefatory items, chapters, sections, tables, figures, appendices, etc.) encountered during the preparation of disquisition using L^AT_EX, therefore the example is relatively elaborate. This example M.S. thesis code shown is included in the file named “`ndsu-example.tex`”. In this example, the examining committee includes the Committee Chair, no Co-Chairs, and only two additional Committee Members. For this example, BIB_TE_X was used to manage references, which would be included in a file named `mybib.bib` separately.

With L^AT_EX, the users type some commands and texts that are specific to their thesis/dissertation, which is human-readable (source code), as shown below following a template, and compile the source to automatically generate the well-formatted NDSU thesis-style document (Fig. 1). The benefits of using L^AT_EX for thesis/dissertation include overall automation, open-source, freely available, vibrant society support, professional quality outcome, elegant mathematics handling, automatic bibliography management, integrated typography principles, portability among operating systems, longer life of the source code, every aspect of document preparation addressed, packages available for specialized needs, thesis/dissertation source code easily converted to journal articles with appropriate templates, and so on.

```
%***** START *****  
\documentclass[ms-thesis,12pt,mathdesign]{ndsu-thesis-2022}
```

```

%Refer documentation (ndsu-thesis-2022-documentation.pdf) for various options and commands

%***** Packages, newcommands, and other customization *****
\usepackage[sort&compress]{natbib}
\citestyle{egu} % agms, agu, arms, egu, cospar, dcu, kluwer, plain, nature
\newcommand{\makebib}{\biblio{apalike}{mybib}} % shortcut for bibliography
\newcommand{\myspacing{1.9}} % 23 lines/page needs 1.9 for thesis

%***** First and second page material *****
\title{The Title of My M.S. Thesis}
\author{Samuel Fargo Bison}
\date{January 2022}
\progdeptchoice{Department} % Use Department (or) Program
\department{Mathematics}

\cchair{Prof. John Adams} % Use actual committee members names
\cmembera{Prof. Abraham Lincoln}
\cmemberb{Prof. George Washington}
\cmemberc{Prof. Theodore Roosevelt} % If 3rd not required - delete this line
\approvaldate{12/14/2022}
\approver{Prof. James Garfield}

%***** Front matter *****
\abstract{This is the abstract for my thesis. \\ \emph{Abstracts for doctoral dissertations must use 350 words
or less. Abstracts for master's papers or master's theses must use 150 words or less.}}

\kant[16]} % dummy text

\acknowledgements{I acknowledge people here. \\ \emph{Acknowledgements text should be placed here.}}

\kant[15]}

\dedication{This thesis is dedicated to my cat, Mr. Fluffles.\\ \emph{This section dedicates the disquisition
to a few significant people. The text must be double-spaced and aligned center to the page.}}
\\ Which is already taken care of by this \LaTeX\ class.}

\preface{You can put a preface here. \\ \emph{This section is optional!}}

\kant[14]}

\listofabbreviations{% may use title case
AC      & Alternating Current \\
NDSU    & North Dakota State University \\
ZL      & Zeta Level % last item does not need \\ but okay to use
}

\listofsymbols{% may use sentence case
$A$     & Area ( $\text{m}^2$ ) \\
$e$     & Euler's constant ( $e$ ) \\
$R^2$   & Coefficient of determination % last item does not need \\ but okay to use
}

%***** Document start *****
\begin{document}
\begin{spacing}{\myspacing} % New line spacing - 23 lines per page

%***** First chapter - paper style *****
\mypaperheading{The First Chapter - Paper Style - Long title of this technical paper}{*}
{This paper is planned to be submitted as a peer-reviewed article \ldots\ more information
about the author(s), title, \emph{journal}, to be added.}

\section{Abstract11}
Paper-styled chapters will have abstracts. Abstract of this chapter goes here. \kant[1]

\section{Section12}
This is the first section of the thesis (1st level: 1.2. Section). \kant[2]

\section{Section13}
This is the second section of the thesis (1st level: 1.3. Section). \kant[3]

\subsection{Subsection131}
This is the subsection text (2nd level: 1.3.1. Subsection). \kant[4]

\subsubsection{Subsubsection1311}
This is the subsection text (3rd level: 1.3.1.1. Subsubsection). \kant[5]

\paragraph{Paragraph13111}
This is the subsection text (4th level: 1.3.1.1.1. Paragraph). \kant[6]

```

```

\section{Table and Figure}
This is the third section of the thesis (1st level: 1.4. Section). This section
illustrates the inclusion of a simple table (\cref{tab:1}) and a figure shown later.

\begin{table}[h]
\centering
\caption{Table captions go at the top of the table. This was long caption of the table
included in the first chapter --- so that we see how it breaks into another line and
having a single spacing. Usually tables are of full-width and are demonstrated subsequently.}
\vspace{-1ex}
\begin{tabular}{c}
\toprule
Number & Month & Days\\
\midrule
\#1 & January & 31\\
\#2 & February & 28\\
\#3 & March & 31\\
\bottomrule
\end{tabular}
\label{tab:1}
\end{table} \kant[7]

Now the figure (\cref{fig:1}) illustrates an example figure from the \texttt{mwe} package.

\myfig[H]{0.525}{example-image-duck}{Caption for this example image in this first chapter.}{fig:1} \kant[8-9]

%***** Second chapter - regular *****
\myheading{The Second Chapter - Regular Style - Long title for this chapter}

Regular style chapters will not have abstracts. General information or outline of the chapter is given
here --- before breaking into sections.

\section{Excellent Results}
This is another section of the thesis (1st level: 2.1. Experimental Results). \Cref{tab:2} presents the
results in a tabular form that spans the entire width. Please note the results shown (\cref{tab:2})
are preliminary.

\begin{table}[ht]
\centering
\caption{Table spanning entire width (full-width) using \texttt{setlength} and
\texttt{tabcolsep}}
\vspace{-1ex}
\setlength{\tabcolsep}{3.75em}
\begin{tabular}{@{\hspace{2ex}} lccr @{\hspace{2ex}} }
\toprule
Number & Name of month & Days & Season\\
\midrule
\#4 & April & 30 & Spring\\
\#5 & May & 31 & Summer\\
\#6 & June & 30 & Summer\\
\bottomrule
\end{tabular}
\begin{tablenotes}[flushleft]
\footnotesize
\item \hspace{-1ex} \emph{[Note]:} The \texttt{tablenotes} environment produces table footnotes.
\end{tablenotes}
\label{tab:2}
\end{table} \kant[7-8]

\subsection{Minor Results}
This is a subsection of the thesis (1st level: 2.2. Experimental Results). \kant[8]
The \Cref{fig:2} is an example image with command showing all arguments including the optional
caption placement. The example figure (\cref{fig:2}) is included in the \texttt{mwe} package.

\myfig[2ex]{H}{0.45}{example-image}{Caption for this example image demonstrating an optional 2ex vertical
spacing. Compare this with a narrow caption spacing without optional argument in \cref{fig:1}.}{fig:2}
\kant[8]

\section{Some References}
Referring to all entries in the “\texttt{mybib.bib}” file to generate the citations here and the listing
using the \texttt{\textbackslash citep{\ldots}} “\texttt{natbib}” command (cite parenthesis) \citep{texbook,
lcompanion,latex2e,knuth1984,leski1977,amsthm2017,calvo2004using,cannayen2011latex,kopka2004guide,notso2021,
bari2016identification}.

The same using \texttt{\textbackslash citet{\ldots}} command (cite text) in the running text as: The authors
\citet{texbook,lcompanion,latex2e,knuth1984,leski1977,amsthm2017,calvo2004using,cannayen2011latex,kopka2004guide,
notso2021,bari2016identification} have something to do with \LaTeX. For most bibliography citations and list
creation, these two commands are sufficient.

```

```

%***** Bibliography handling *****
\makebib % Shortcut command using style and bib defined in the preamble

%***** Named appendix A *****
\namedappendices{A}{Named first appendix}
Appendix material can be included here. First including a figure (fig.~\ref{fig:ap1}).

\section{Appendix A - Section With Figure}
\myfigap{H}{0.5}{example-image-golden}{A golden ratio rectangle image.}{fig:ap1} \kant[8]

\section{Appendix A - Section With Table}
And, then including a table (table.~\ref{tab:ap1}).

\begin{appendixtable}[h!]
\centering
\caption{Use of \texttt{tblr} environment for full-width table - applicable to both main text and appendix.
Note the use of \texttt{booktabs} commands and 'X' parameters to reproduce Table~\ref{tab:2}.}
\begin{tblr}{*4X}
\toprule
Number & Name of month & Days & Season\\
\midrule
\#7 & July & 30 & Spring\\
\multicolumn{4}{c}{The three columns combined} \\
\#8 & August & 31 & Summer\\
\#9 & September & 30 & Summer\\
\bottomrule
\end{tblr}
\begin{tablenotes}[flushleft]
\footnotesize
\item \hspace{-1ex} \emph{Note}: The \texttt{tablenotes} environment produces table footnotes.
Refer to \texttt{tabularray} documentation for further details.
\end{tablenotes}
\label{tab:ap1}
\end{appendixtable}

\subsection{Appendix A Subsection}
\kant[10]

%***** Named Appendix B *****
\namedappendices{B}{Named second appendix}
Appendix material can be included here. First including a figure (fig.~\ref{fig:ap2}).

\section{Appendix B - Section With Figure}
\kant[9]
\myfigap[0.5ex]{H}{0.6}{example-grid-100x100pt}{A $10 \times 10$ grid of different concentric colors.}{fig:ap2}

\section{Appendix B - Section With Table}
Now coding another appendix table (table.~\ref{tab:ap2}) that spans the entire width using the manual method
(using 'tabcolsep' command; and 'resize' command to fit large tables).

\begin{appendixtable}[h]
\centering
\caption{Squares and cubes named appendix table using \texttt{siunitx} and \texttt{tabularray}
packages.}
\begin{tblr}{X X[c] X[r] X[1.5,r]}
\toprule
Number & Square & Cubes & Fourth power\\
\midrule
11 & 121 & \num{1331} & \num{14641}\\
22 & 484 & \num{10648} & \num{234256}\\
333 & \num{110889} & \num{36926037} & \num{12296370321}\\
\bottomrule
\end{tblr}
\label{tab:ap2}
\end{appendixtable}

\subsection{Appendix B Subsection}
\kant[11]

\closeappendices % Automatic LOAT and LOAF generation; remove or comment if not required

\end{spacing}
\end{document}
%***** END *****

```

The example thesis code when compiled will produce the output shown in Figure 1. The example source code serves as a lean template with meaningful and dummy text. The citations in the text and reference

listing were automatically generated according to the selected reference style.

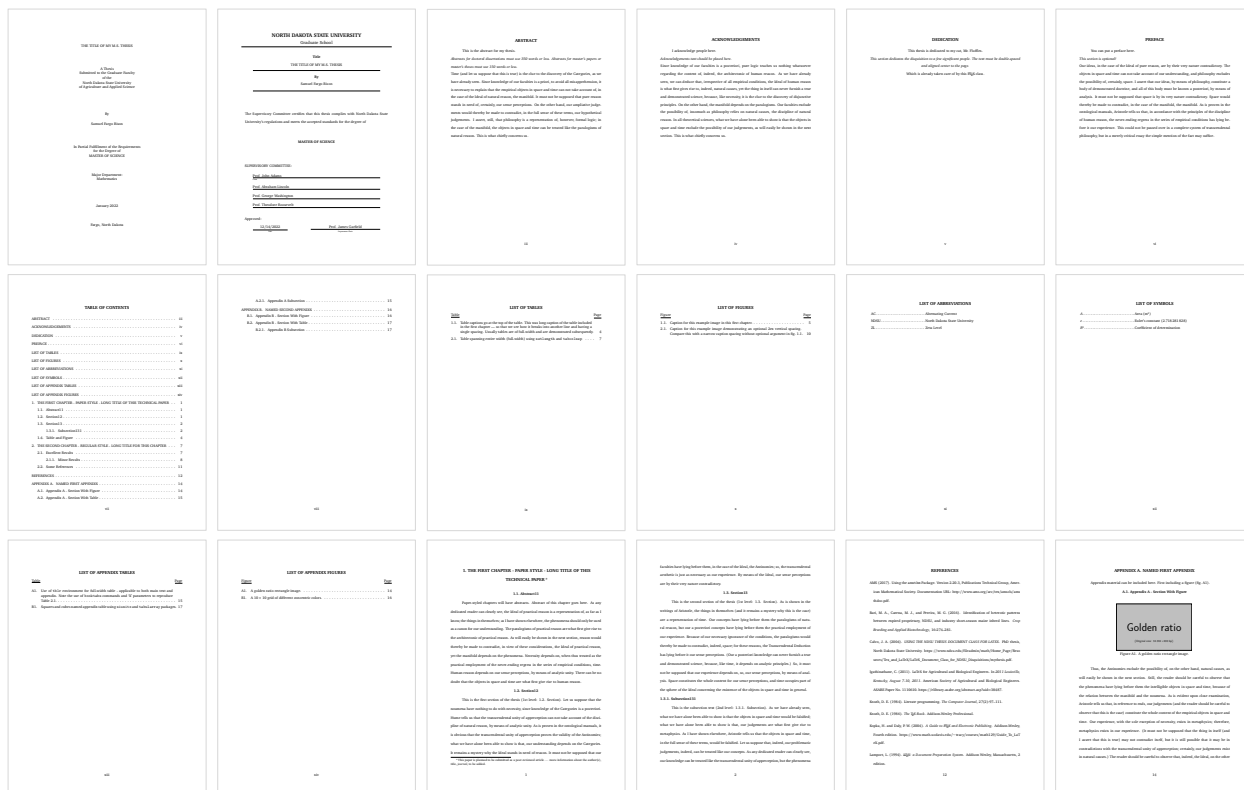


Figure 1: Automatically formatted output sample pages of the example thesis according to NDSU Graduate School requirements. Several pages were skipped to show the overall outcome and the source code.

A lightweight source file named “`ndsu-sandbox.tex`,” which can be used to try out things conveniently in the actual NDSU thesis environment was also included in the package folder. Things tested here (including the bibliography) can be readily inserted into the original thesis/dissertation document. In addition, another extended file named “`NDSU-Thesis-Extended.tex`” containing several additional comments (listing the various options of the class) and features was made available as a supporting document. The code below (`ndsu-example.tex`) will also work when directly extracted (selected and copied) and compiled. Most of the requirements of the students for disquisition will be covered with these two example source code files.

4 Documentclass Options

These are the options passed to the `documentclass` command while calling the class. These options essentially affect the whole document and a default behavior (no options specified, shown below) was also valid.

```
\documentclass{ndsu-thesis-2022}
```

The above default behavior with no [options] specified produces a Ph.D. dissertation in 12 pt font size with auto-numbered heading and justified text in computer modern font. However the command:

```
\documentclass[ms-thesis,11pt,nonumber,nojustify,draft,showframe,times]{ndsu-thesis-2022}
```

produces an M.S thesis in 11 pt font size unjustified paragraphs text with unnumbered heading in draft mode in Times Roman font and shows the frame using the set margins. The order in which these options are passed does not matter. The various options are briefly discussed subsequently.

4.1 Disquisition degree and type

By default, this class assumes the document is a Ph.D. dissertation. Providing a degree option will accommodate other available degree and disquisition types (Table 1):

Table 1: Options for degree and disquisition types

Option	Degree	Disquisition type
[phd]	Ph.D.	Dissertation
[ms-thesis]	M.S.	Thesis
[ms-paper]	M.S.	Paper
[ma-thesis]	M.A.	Thesis
[ma-paper]	M.A.	Paper

4.2 Font size

The general font sizes used with thesis are 10, 11, and 12 points and they vary with the selected font. The available options (any of these used) are:

```
\documentclass[10pt (or) 11pt (or) 12pt]{ndsu-thesis-2022}
```

The default was set as 12pt.

4.3 Auto-numbered, chapter-numbered, and unnumbered styles

The three possible NDSU thesis styles with options included are: (i) Auto-numbered [default option] — where chapters, sections, subsections, and so on will be numbered; (ii) Chapter-numbered [chapternumber] — where only chapters are numbered, while sections, subsections, and so on will not be numbered; and (iii) Unnumbered [nonumber] — where all headings such as chapters, sections, subsections, and so on will not be numbered.

As the default is the numbered style, the chapter-numbered and unnumbered styles were produced by the “chapternumber” and “nonumber” options respectively as:

```
\documentclass[chapternumber (or) nonumber]{ndsu-thesis-2022}
```

The default was the “Auto-numbered” style. These options will have their specific effect on the numbering scheme of the tables and figures.

4.4 Paragraph text justification

Based on their preference students can follow fully-justified (with hyphenated words and word wrapping) or unjustified (no word breaking but right margin ragged, aka left-justified). As the default is justified, the left-justified passages were produced by the [nojustify] option. For justified style, nothing needs to be specified. NDSU approves both styles.

4.5 Draft and display document frames

You can use the [draft] option to place the disquisition into draft mode. In this mode, margin overflows are marked with a heavy black box to draw your attention to them; additionally, images are replaced by a placeholder (see Fig. 2). If you import other packages in your disquisition, they may also change their behavior when in draft mode.

The [showframe] option (based on geometry package) produces a frame around the text area which can be used to check how the text aligns with the margins (left, right, top, and bottom; see figure above). The illustration alongside displays the result of these options showing the overflowing text, bottom margin frame, right margin frame, margin notes frames, and the overflow heavy black box. The default behavior is these options were inactive.

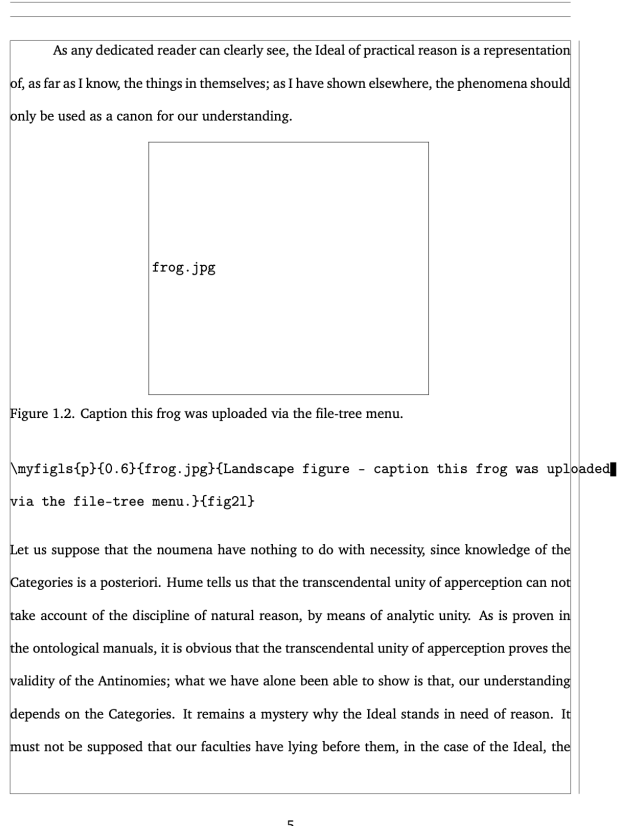


Figure 2: Use of `draft` and `showframe` options in `documentclass` producing image placeholder for quicker processing, document frames, and margin overflows.

4.6 Fonts

The following font options [`bookman`, `charter`, `gentium`, `kpfonts`, `libertine`, `mathdesign`, `mathptmx`, `newcent`, `palatino`, `tgtermes`, `times`, `tgbonum`, `tgpagella`, `tgschola`, `utopia`, `zlmmtt`] are compatible with the class, and anyone can be used. The default was L^AT_EX computer modern font. Users are urged to check the NDSU-approved fonts and select those that resemble them and use them with appropriate font sizes.

4.7 Individual or combine reference listing

The two types of reference listings are in the individual chapters added that end or a combined reference listing for all chapters as a standalone unnumbered chapter.

5 Preamble Information

5.1 General information - packages and shortcuts

If your disquisition requires the use of additional L^AT_EX packages, macro files, or other commands, include them in the preamble. Packages such as `natbib` (author-year and number style citations) or other competent reference handling packages and their options can be loaded. Similarly, mathematical theorem environment-related commands (theorem, corollary, lemma) through `\newtheorem` of `amsthm` package, `caption` package

setups through `\captionsetup[type]{options}`, where *type* = table, figure, or subfigure, and other shortcuts for repetitive longer commands or text can be defined in the preamble. As these are specific to the users and the requirements vary with the users of different specializations, these were not included in the class. Therefore suitable packages, commands, and shortcuts can be defined by the users.

5.2 Dissertation front pages

Before issuing the `\begin{document}` command, several pieces of preamble information are available.

5.2.1 Title

Include the title of the disquisition using the `\title{...}` command. This is required.

5.2.2 Author

Include the full name of the disquisition author using the `\author{...}` command. This is required.

5.2.3 Major department/program choice

Specify whether it is “Department” or “Program” that is applicable using the `\progdeptchoice{...}` command. This will produce ultimately “Major Department:” or “Major Program:” based on the input choice. This is required.

5.2.4 Department or program

Include the name of the major department or program using the `\department{...}` command. This is required.

5.2.5 Degree option

If the major department or program has a degree option, indicate this using the `\degreeoption{...}` command. This is optional.

5.2.6 Date

Include the date of the final examination using the `\date{...}` command. The accepted format of this date is *month year* as: `\date{October 2022}`. This is required.

5.2.7 Examining committee

Include the Chair (or Co-Chairs) and members of the examining committee using separate commands. The `\cchair{...}` command is used to indicate the committee Chair. Use `\cochairZ{...}` to indicate any committee Co-Chair members, where Z is a or b. This class does not support more than two Co-Chairs and four Committee Members. Use the `\cmemberX{...}` to indicate other committee members, where X is a, b, c, or d. Use only as many of these commands as needed to list all committee members.

5.2.8 Approval information

Use the `\approvaldate{...}` command to include the full date of disquisition approval (i.e. month/date/year). This date is generally the date the thesis was approved by the Department Chair following the defense after the approval (usually electronically) of all committee members. Use `\approver{...}` to include the Department Chair who approved the disquisition. Both commands are required.

5.3 Dissertation front matter

5.3.1 Abstract

Use the `\abstract{...}` command to include the disquisition abstract. Abstracts for doctoral dissertations must use 350 words or less. Abstracts for master’s papers or master’s theses must use 150 words or less. This is required.

5.3.2 Acknowledgements

If the disquisition includes acknowledgements, include them using the `\acknowledgements{...}` command. This is optional.

5.3.3 Dedication

If the disquisition includes a dedication, include it using the `\dedication{...}` command. This is optional.

5.3.4 Preface

If the disquisition includes a preface, include it using the `\preface{...}` command. This is optional. The NDSU guidelines state:

“The Preface can provide an autobiographical account of how the disquisition came to be or include a significant quote that drove your research. Follow the General Requirements for font, spacing, and page numbers for prefatory materials.”

6 Automatic Components

Several automatic components will be generated, as a part of the front matter, based on the source code of the dissertation and are briefly described. Based on the department, requirement, and style of the thesis some of the items such as, lists of abbreviations, symbols, and appendix tables and figures (Secs. 6.3–6.5) may be dropped from the coding.

6.1 Table of contents

The table of contents (TOC) gets automatically generated with entries up to three levels sections (`\my...heading`, `\section`, and `\subsection`). The dissertation may have further levels of sections but they are not shown in the TOC.

6.2 List of tables and figures

The list of tables (LOT) and list of figures (LOF) will be generated based on the table and figure full captions in the `table` and `figure` environments. New commands for handling figures such as `\myfig{1+5 arguments}`, and `\myfigls{1+5 arguments}` with their own `[optional]` argument to adjust the position of the caption with respect to figure element were defined.

6.3 List of abbreviations

The collection of abbreviations used in the dissertation can be made in to a list of abbreviations (LOA) using the `\listofabbreviations{...}` command. This collection should be alphabetized before coding. This will be a two-column tabular entry. A two entry example of LOA code and the output are shown below:

```
\listofabbreviations{
AC & Alternating current\\
NDSU & North Dakota State University}
```

LIST OF ABBREVIATIONS

```
AC..... Alternating current
NDSU..... North Dakota State University
```

6.4 List of symbols

The collection of all technical symbols used in the dissertation, usually coded in “math” mode, can be made into a list of symbols (LOS) using the `\listofsymbols{...}` command. This collection should be alphabetized before coding and math mode should be used as required. This will be a two-column tabular entry. A three-entry example of LOS also using the `siunitx` package and the output are shown below:

<code>\listofsymbols{</code>	
<code>\$A\$ & Area (m^2)</code>	LIST OF SYMBOLS
<code>\$c\$ & Speed of light ($299.792 \text{ km s}^{-1}$)</code>	AArea (m^2)
<code>\$R^2\$ & Coefficient of determination</code>	cSpeed of light ($299.792 \text{ km s}^{-1}$)
	R^2Coefficient of determination

6.5 List of appendix tables and figures

The list of appendix tables (LOAT) and list of appendix figures (LOAF) will be generated based on the appendix table and figure full captions in the `appendixable` and `appendixfigure` environments. New commands such as `\myfigap{5 arguments}` and `\myfigapls{5 arguments}` plus one [optional] argument for adjusting the caption placement for regular and landscape figures were defined.

The `\closeappendices` command should be issued at the end of the last appendix, which ensures the automatic creation of the LOAT and LOAF when the appendices had tables and figures. If the appendices had only tables or only figures then the commands `\closeappendixtables` or `\closeappendixfigures` should be used to avoid blank entries, and if no tables are figures used in appendices then these commands should not be used or commented out.

7 Beginning the Document

After including the necessary preamble information, use `\begin{document}` to start the document. This command automatically generates the necessary cover pages and other automatic components. The usual `\maketitle` command should not be used, as it was already issued in the class.

8 Headings

Major headings (e.g. chapters) are issued using the `\myheading{...}` command. This command supersedes the usual `\chapter` command, which should not be used. The following shows the hierarchy of headings:

<code>\myheading{...}</code>	Produces all-caps chapter headings automatically - 0th level
<code>\mypaperheading{...}</code>	Produces all-caps paper chapter headings with footnote - 0th level
<code>\section{...}</code>	Produces centered, bold headings (use title case) - 1st level
<code>\subsection{...}</code>	Produces left-aligned, bold headings (use title case) - 2nd level
<code>\subsubsection{...}</code>	Produces left-aligned, bold, italic headings (use sentence case) - 3rd level
<code>\paragraph{...}</code>	Produces left-aligned, italic headings (use sentence case) - 4th level

Each `\myheading{...}` or `\mypaperheading{3 args}` command starts a new page and entry in the table of contents. The regular chapter command is simple and takes one argument which is the title of the chapter as `\myheading{title}`. The paper-styled chapter takes three arguments that address the title and footnote as `\mypaperheading{title}{footnotemark}{footnotetext}`. The `title` is common to both styles and will be rendered as all-caps irrespective of the input. The `footnotemark` usually be an asterisk (*), but can be any valid footnote mark including math symbols. The footnote mark will be used in the footnote text but will not feature in the TOC. The footnote text will be rendered as a footnote on the same page and automatically uses the input footnote mark.

In general, the paper-styled chapter requires an “Abstract” section (title coded as a section), while the regular chapter does not. The class is coded to produce a consistent space between the title and the text

(or section) below the title; however when necessary `\vspace{+ve or -ve}` can be issued before the plain introductory text or section command to adjust this vertical space.

Instances of `\subsubsection{...}` and `\paragraph{...}` do not appear in the TOC, though they are included in the document. Other than the chapter headings, the rest of the item headings should be coded by the user manually with appropriate capitalization (title and sentence cases).

9 Dummy Text and Images

Users will be curious to see what their thesis/dissertation will look like quickly without using the actual texts and figures. The class comes loaded with necessary packages such as `kantlipsum` (for dummy text — philosophical prose paragraphs in English) and `mwe` (“minimal working examples” for dummy images). These will help visualize the whole document (fonts, spacings, and layout) with minimal effort, and this is a common practice among typesetters to use such dummy text and images. Commands from these packages are used in the thesis example (Sec. 3).

Commands like `\kant[1]` or `\kant[4-8]` will produce single or multiple dummy text paragraphs. Similarly, dummy images included in the `mwe` package can be accessed using their specific names and can be used as the image argument in the `\includegraphics` command, which means that the user need not use their images. Some of the commonly used examples images are: `example-image`, `example-image-a`, `example-image-b`, `example-image-c`, `example-image-16x10`, `example-image-golden`, `example-image-plain`, `example-image-duck`, `example-image-empty`, and `example-grid-100x100pt`. Refer to the documentation of these packages for further information.

10 Tables and Figures

10.1 Tables

Different kinds of tables, such as simple table without caption (`tabular`), table with caption (`table`), table with footnote (`threeparttable`), table spanning entire text width (`tblr`), table spanning multiple pages (`longtable`), and table in landscape page (`pdflscape`) can be coded following the documentation of respective packages, and no shortcuts were defined as they were not practical. Using `booktabs` package professional quality tables (Sec. 4.1) can be created.

10.1.1 Full-width tables — `tblr` environment from `tabularray` package

Full-width tables are suggested for NDSU theses/dissertations. A quick and efficient method of creating tables that automatically span the entire textwidth is the use of `tblr` (short for `top-bottom-left-right` or `tabularray`) environment instead of the usual `tabular` inside the `table` environment. The `tblr` environment (from `tabularray` package) uses a special column justification code `X` (default & options). This `X` code allots fixed column width based on the number of columns specified (default) and customizes individual columns’ proportional width using coefficients with options. The `tblr` also takes the usual `l`, `c`, `r`, and `p` justification codes as well as the commands of `booktab` in the usual manner. The `tblr` environment is from the latest package and has several special features such as, specifying exclusive `math` mode column (no `$` symbols required for individual items); SI units features; colored rows, cells, and lines; and so on (See `tabularray` package documentation).

10.1.2 Full-width tables — manual method

However, for the best control of tables, especially with more columns, a combination of `\resizebox` (resizing the entire table - mostly for scaling down) and `\tabcolsep` (maintaining the column separation space) works the best. Thus, the command `\resizebox{\columnwidth}{!}` makes the table to span the entire text width of the page. This will expand or shrink the contents of the table to fit the entire width. It should be fine with the fonts shrink to fit the width, but will not be when the fonts enlarge (especially when the table

is small and has only a few columns). In such situations, the space between the columns can be adjusted using the `\tabcolsep{...}` command, where increased spacing reduces the font size and *vice versa*. Thus, by using these commands (including `tblr`) in combination the tables and the font size can be scaled down to fit the page with the proper font size.

10.2 Figures

It should be noted that the manual coding of figures using “**figure**” environment with `\includegraphics{...}` centering, resize, caption and labels is the direct approach.

10.2.1 Shortcuts for figures — direct and optional

However, for convenience, a set of single command shortcuts, with five arguments plus one optional argument are defined. These commands specify (1) [optional] vertical placement of the caption (moving it up and down with respect to the bottom of the figure, especially for images with excessive or too less whitespace), (2) placement, (3) size factor, (4) input file, (5) caption, and (6) label were defined to produce figures (regular and landscape). The default caption’s `aboveskip` is `0ex`, and this value can be changed using the optional argument. Following are the examples of figure shortcuts for regular and landscape figures without and with the optional argument. These shortcuts are automatically included in the LOF and LOT that appear after the TOC.

```
\myfig{ht}{0.7}{image1.jpg}{Caption for this regular figure}{fig:1}
\myfig[1.5ex]{ht}{0.7}{image1o.jpg}{Figure caption with placement option}{fig:1o}

\myfigls{p}{1.32}{image2.pdf}{Caption for this landscape figure}{fig:2ls}
\myfigls[2ex]{p}{1.31}{image3.pdf}{Landscape figure caption with placement option}{fig:3ls}
```

Sometimes, excessive spaces were observed above and below the figures and tables (floating elements) as well as equations with respect to the text around. The use of vertical spacing (+ve or -ve; e.g., `\vspace{4pt}` and `\vspace{-6pt}`) around the floating elements can help in the adjustment of their placements. The vertical spacing commands can be issued before and after these environments (as required) to fix the spacing. Coding tables and figures will automatically create the LOT and LOF. A similar approach can be used for appendix figures (Sec. 13.3).

10.3 Captions

Because of the way spacing is handled, captions in `table` environments must appear at the top of the table, while captions in `figure` environments must appear at the bottom of the figure. If you use both `\caption` and `\label` commands in these environments, the `\caption` command must come before the `\label` command to ensure the environment is numbered correctly. The captions are coded in such a way that shorter ones are centered and longer ones are left-justified; however, as default the table captions are left-justified and can be changed, as outlined subsequently, to fit the requirement. The style of the caption can be basic or specific to the department, usually following the parent technical society’s leading journal. The style of labeling (e.g., regular *vs* bold *vs* italic, naming: Fig. *vs* fig. *vs* Figure, etc.) can also be adopted from the leading journal. The various options available for caption using the `caption` package (already loaded) can be set through the `\captionsetup{...}` command. The common options are `position`, `skip`, `belowskip`, `aboveskip`, `font`, `labelfont`, `labelsep`, `singlelinecheck`, `format`, `justification`, and so on.

11 Equations

About handling equations, the NDSU’s guidelines state “When coding equations, the guidelines call for the equation to be center-aligned, with the equation number aligned flush with the right margin.” The strong

suit of L^AT_EX is the professional manner it typesets the equations and mathematical elements. Show below is the distance formula that was defined and referred (eq. 1), which satisfies NDSU’s guidelines:

$$\text{Distance formula: } d = \sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2} \quad (1)$$

where, d is the distance; and x_1, y_1, x_2 and y_2 are the coordinates of the two points. The equations can be displayed (e.g., eq. 1) produced by `$$...$$` or `\[...\]` or `equation` environment; and the inline as: $ax^2 + bx + c = 0$ produced by `$....$` or `\(...\)`. There exists several other commands are available to produce the equations through several packages (e.g., `align`, `array`, `eqnarray`, `gather`, `split`) and any imaginable mathematical information can be coded. Also L^AT_EX supports a huge list of symbols (Refer: The Comprehensive L^AT_EX Symbol List; 18,150 symbols; 449 pages) that can be used in general text or equations.

Use of subscript and text in equations - need to do this

12 References/Bibliography

The two most common bibliography management systems are BibTeX and BibLaTeX; the former being simpler and the latter being modern and highly versatile. Reference or bibliography chapter or section can be combined into a stand-alone chapter (whole) or the reference listing can be included in all individual chapters. The bibliography listing in individual chapters sometimes desired by the user can be easily coded using the advanced BibL^AT_EX, which is also coded in the class. Both systems (individual or whole) use the same reference data in the form of `*.bib` file. The authors recommend using any of the systems by appropriately employing the system-specific commands. For bibliography compilation issues see Sec. 14.7.

12.1 Using BibTeX

12.1.1 Cite while you write (CWYW) using natbib

The `natbib` package for bibliography management is widely used and very stable and follows the CWYW paradigm. The package produces both author-year and numerical citations. The commands like `\citep{...}` citation in parenthesis and `\citet{...}` citation in running text are quite useful in particular. These commands will produce the following outputs, for example: “(Author et al., 2022)” and “Author et al. (2022) found ...”. The compatible styles (`*.bst`) with `natbib` and NDSU class are that work with standard L^AT_EX installation are: `abbrvnat`, `agms`, `agu`, `apalike`, `apalike2`, `authordate1`, `authordate3`, `cell`, `chicago`, `chicagoa`, `dcu`, `dinat`, `IEEEtran` (family; numerical styles), `kluwer`, `plainnat`, `rusnat`, `unsrtnat`, and so on. For other styles, users can able to download the specific style (`*.bst`) files and have them in the local folder. As `natbib` is an optional bibliography system, it was not coded in the class, and to use `natbib` the following code should be in the preamble:

```
\usepackage[sort&compress]{natbib}
\citestyle{arms} % agms, agu, arms, egu, cospar, dcu, kluwer, plain, nature
```

It is convenient to load the `natbib` package with minimal options, as shown above, and choose predefined `\citestyle{option}` options producing several styles defined in the package.

12.1.2 Bibliography generation and files handling

The two basic commands that is required to implement BibTeX system are:

```
\bibliographystyle{style} % See list of styles (Sec.11.1.1)
\bibliography{name-of-bib-file}
```

However, a single new command “`\makebib`” (direct shortcut with no arguments) developed, replacing the above commands, sets up the bibliography style and `*.bib` file arguments that are being utilized in other shortcuts is:

```
\newcommand{\makebib}{\biblio{style}{name-of-bib-file}}
```

Now, simply issuing the `\makebib` will generate the references listing based on the style and bib file input in the above command. The “style” of bibliography (`*.bst`) entries (typically `plainnat` or `apalike`), is controlled by the first argument; the user is referred to the BibTeX manual for formatting details and other available styles, such as those provided by the peer-reviewed journals related to the specialization. The “name” used in the second argument must be the same as the name of the bibliography (`*.bib`) file, but with the extension removed. Once correct citation commands (`\citep{...}` and/or `\citett{...}`) are issued following CWYW, the citation with proper reference number or entry will appear in the text and listings in the proper style (based on `*.bst`) will be generated. The above shortcut generates an unnumbered chapter with the title REFERENCES (accepted by NDSU) and also a corresponding TOC entry.

These commands (or equivalent commands if the user uses a different bibliography management system) are optional but are required if the disquisition includes references. Basic bibliography citation command is `\cite{...}`.

12.2 Using Bib_{La}TeX

The Bib_{La}TeX package provides advanced bibliographic facilities for use with LaTeX. Good working knowledge in LaTeX should be sufficient to design new bibliography and citation styles using this system. The Bib_{La}TeX works with the backend (program) “biber”, which is used to process the bibliography data files and then performs all sorting, label generation, and many more operations. This package also supports subdivided bibliographies, multiple bibliographies within one document, customizable sorting, multiple bibliographies with different sorting, customizable labels and bibliographies may be subdivided into parts and/or segmented by topics. Users are urged to refer to the package documentation for various features (<https://ctan.org/pkg/biblatex?lang=en>).

12.2.1 Commands, cite, bibliography generation and files handling

The basic commands that invoke the Bib_{La}TeX system, which has to be issued in the preamble, are:

```
\usepackage[style=apa,natbib=true,backend=biber]{biblatex}
\addbibresource{name-of-bib-file.bib}
```

In the above Bib_{La}TeX command’s option the bibliography information processing program “biber” was used as a backend program. The options also loads “apa” style and “natbib” handling (allowing the `\citep{...}` and `\citett{...}` commands in Bib_{La}TeX) as an example. The documentation and other resources (<http://tug.ctan.org/info/biblatex-cheatsheet/biblatex-cheatsheet.pdf>) may be referred for common options and details of the package. The compatible styles used with Bib_{La}TeX are: `numeric`, `numeric-comp`, `alphabetic`, `authoryear`, `authoryear-icomp`, `authortitle`, `verbose`, `reading`, `draft`, `apa`, `chem-acs`, `chem-angew`, `chem-biochem`, `chem-rsc`, `ieee`, `mila`, `musuos`, `nutre`, `nejm`, `phys`, `science`, and `oscola`. Users can use an appropriate style to match their specialization style guide.

With the package and bib file(s) loaded and processed, the reference listing can be generated anywhere in the document by issuing:

```
\printbibliography[heading=bibintoc,title=REFERENCES]
```

The options “`heading=bibintoc`” makes an unnumbered chapter and includes the heading in the TOC and “`title=REFERENCES`” changes the default title from BIBLIOGRAPHY to REFERENCES. The `\printbibliography` command when issued at the end of the chapters will create a “combined” REFERENCES chapter.

12.2.2 Individual chapters bibliography (multiple)

Sometimes it is desired to have a bibliography listing in every chapter, as the last unnumbered section, especially with the paper-style chapters. Chapters bibliography can be easily processed using Bib_{La}TeX than using BibTeX. For the bibliography listings only drawn from individual chapters they have to be restricted in the `refsection` environment as:

```

\begin{refsection}
...Chapter's text starting with abstract, sections/subsections, and so on,
with citations using \cite{...} and other citation commands ...
\printbibliography[heading=subbibintoc,title={References}]
\end{refsection}

```

The options of the `\printbibliography` command such as “`heading=subbibintoc`” will make the bibliography as unnumbered section and also creates a corresponding TOC entry, while “`title=References`” renames the default title.

12.3 Specialization specific bibliography — ASABE and IEEE examples

The bibliography styles to be followed in thesis/dissertation will be based on the parent department and the major technical society the department subscribes to in general. For example, the NDUS’s “Agricultural and Biosystems Engineering” department’s major technical society is The American Society of Agricultural and Biological Engineers (ASABE), and the NDSU’s “Electrical and Computer Engineering” department’s technical society is The Institute of Electrical and Electronics Engineers (IEEE). The two ways of making the bibliography are to use BibTeX or BibL^AT_EX.

The ASABE style journal references were similar to `apalike2` style but they use & before the last author instead of “and”. Therefore a modified style that achieves this called `myASABE.bst` was developed and included to produce the desired outputs (however, BibL^AT_EX with `style=apa` produces the & directly). For other commands, the `natbib` package documentation should be referred to.

For IEEEtran family of styles (`-N`, `-S`, `-SA`, `-SN`) that are numeric bibliography styles, use option “plain” in “`citestyle`” command as `\citestyle{plain}`. It is also possible to load the package with the options directly as well in one command.

Option 1

```

\usepackage[numbers,sort&compress]{natbib}
\bibliographystyle{IEEEtranN}%IEEEtranN

```

Option 2

```

\usepackage[sort&compress]{natbib}
\citestyle{plain}
\bibliographystyle{IEEEtranN}%IEEEtranN

```

Option 3

```

% Need to spell out authors
\usepackage[numbers,sort&compress]{natbib}
\bibliographystyle{IEEEtran}%IEEEtran

```

Option 4

```

\usepackage[sort&compress]{natbib}
\citestyle{plain}
\bibliographystyle{IEEEtran}%IEEEtran

```

Option 5

```

\bibliographystyle{IEEEtran}%IEEEtran

```


13 Appendix

13.1 Single and multiple named appendices

If the disquisition includes a single appendix or multiple named appendices, one of two commands must be used to produce them. If the dissertation has only one appendix, use the `\appendix` command to begin it. This command generates an un-lettered APPENDIX chapter that can have sections, subsections, and so on, as well as tables, figures, and other elements.

If multiple named appendices are necessary, use the `\namedappendices{...}{...}` that can also contain other elements. Following are two examples of the named appendices:

```
\namedappendices}{A}{First named appendix title here}  
\namedappendices}{B}{Second named appendix title here}
```

These appendix commands are optional but are required if the disquisition includes an appendix. The appendix must follow the unnumbered REFERENCES chapter. NDSU's guidelines on appendices only allow named appendices with letters (e.g., "APPENDIX A", "APPENDIX B"), while numerical or other styles ("APPENDIX 1", "APPENDIX 2", "APPENDIX I", "APPENDIX II", and so forth) are not accepted.

It is necessary to generate the listing of appendices in TOC up to subsection level (A.1.1), similar to the regular chapters. To achieve this necessary codes were included in the class.

13.2 Appendix figures and tables

If the appendix contains figures or tables, use the `appendixfigure` and `appendixtable` environments to generate them. These special environments ensure that the figures and tables appear in separate tables that appear after the table of contents. The usual `figure` and `table` environments should not be used in the appendix. The same rules for centering, captions, and labels used in normal `figure` and `table` environments apply to `appendixfigure` and `appendixtable` environments. It should be noted that the manual coding of `appendixfigure` is also possible with basic commands.

Similar to figures handled in the regular chapters (Sec. 10.2), for appendix figures as well, single command shortcuts dealing with appendix figures and appendix landscape figures with necessary arguments and one optional argument for caption vertical placement, were defined to produce the figures. Following are the examples of figure shortcuts for appendix regular, and landscape figures:

```
\myfigap{H}{0.6}{image_ap1.jpg}{Caption for this appendix regular figure}{fig:ap1}  
\myfigap[12mm]{H}{0.6}{image_ap2.jpg}{Appendix landscape figure caption}{fig:ap2}  
  
\myfigapls{p}{1.32}{image_ap3.pdf}{Caption for this appendix landscape figure}{f:ap3ls}  
\myfigapls[1.5ex]{p}{1.33}{image_ap4.pdf}{Appendix landscape figure caption}{f:ap4ls}
```

13.3 Closing appendices and creating TOC, LOAT, and LOAF

In the class, the creation of LOAT and LOAF as well as their TOC entries requires special consideration. When the last appendix has got at least one table and one figure, the TOC, LOAT, and LOAF will be automatically generated without intervention. However, when the last appendix does not have at least a table, or figure, or both (even though the previous appendices had them) the corresponding LOAT or LOAF and relevant TOC entries will not appear. The solution for the automatic creation of these items, irrespective of the contents of the last appendix is issuing the command `\closeappendices` in the overall code somewhere before `\end{document}` and after the last appendix code.

When the last appendix had both table(s) and figure(s), then issuing this command is not necessary but okay to use it anyhow for completeness. Furthermore, if the other appendices had only tables and the last does not then use `\closeappendixtables`, similarly for figures then use `\closeappendixfigures` for exclusive creation of LOAT or LOAF with relevant TOC (See Sec. 6.5). In case when a thesis had no tables or figures in the appendices or no appendices in the thesis, then the aforementioned `\closeappendi...` commands should not be issued or commented, to avoid the generation of LOAT, LOAF, and TOC entries.

14 Additional Information I — Special Commands

14.1 Whole document text spacing

NDSU mandates double-spacing for the body paragraphs text. A default double-spacing setting in MS Word produces 23 lines per page while L^AT_EX `\doublespacing` produces 27. Any of these lines per page defaults in the respective systems are acceptable. To recreate the line spacing of 23 lines per page in L^AT_EX was produced by defining `\newcommand\myspacing{1.9}` in the preamble and issuing:

```
\begin{spacing}{\myspacing} ... All other text ... \end{spacing}
```

This `spacing` command should immediately follow `\begin{document}` and closed just before the `\end{document}`. Other values of `\myspacing` will produce other spacings. The `\myspacing` was not used in the “Thesis Example” (Sec. 3) and the class by default produces 27 lines per page.

However, for the front matter, the `spacing` environment should be enclosed within the front matter component. For example, the abstract with different spacing (than the default 27 lines per page) should be coded as:

```
\abstract{ \begin{spacing}{\myspacing} ... Abstract text ... \end{spacing}}
```

A similar approach should be followed for other components such as acknowledgments, dedication, and preface.

14.2 Tables: Advanced commands

14.2.1 Table row spacing and fonts

The row spacing can be adjusted, if desired, using `\renewcommand{\arraystretch}{...}` inside the table environment. For example, a value of 1.75 for the `arraystretch` will be similar to the double line spacing; and without this command, the row spacing will be single line spacing. Table footnotes can be added through `tablenotes` environment placed inside `table` environment after the `tabular` and `resize` blocks. The font size can be altered by selecting the standard sizes (e.g., `\footnotesize`, `\small`) within the `tablenotes` environment.

14.2.2 Landscape tables

When a table has more columns of information, the most common solution is the landscape orientation which is achieved through `landscape` environment by enclosing the `table` codes (which may contain other elements) inside `landscape` environment block (between `\begin{landscape}` and `\end{landscape}`). With `landscape` usually the placement option will be `[p]` and the whole width should be set around 1.32 times the `\columnwidth`, or adjusted suitably to leave acceptable margins all around.

14.2.3 Long tables

To do When a table has more columns of information, the most common solution.

14.2.4 Landscape long tables

Sometimes, there will be a need to create long tables (multiple pages spanning tables) in the landscape orientation to accommodate several columns that will not fit in regular paper orientation. This landscape long tables can be logically obtained by enclosing the long table codes within `landscape` environment block as described earlier. It should be noted that, unlike regular tables, long table source code involves several components (main caption, running header [abbreviated caption], running footer [usually the word “continued...”], and main footnote). Refer the “NDSU-Thesis-Extended.tex” for an example code.

14.3 Figures: Advanced commands

14.3.1 Figures in separate folder

Several images (graphs, drawings, and pictures) were used while developing a thesis or paper. It will be convenient to store all these images in a subfolder to reduce the clutter. The following command should be issued in the preamble indicating the name of the subfolder (e.g., `figures`) relative to the main `.tex` file as:

```
\graphicspath{{./figures/}}
```

The type of image files applicable are: `jpg`, `pdf`, `png`, and `eps`. It is also possible to give an absolute path to the images folder in the above command.

14.3.2 Flowchart - tikz package

Flowcharts, schemes, geometrical diagrams, circuit diagrams, and data visualization graphs are common in technical writing. These elements can be created elsewhere and included in the dissertation as an image or high-quality (vector graphics) can be created using codes directly. An example of a flowchart created through TikZ code is shown below:

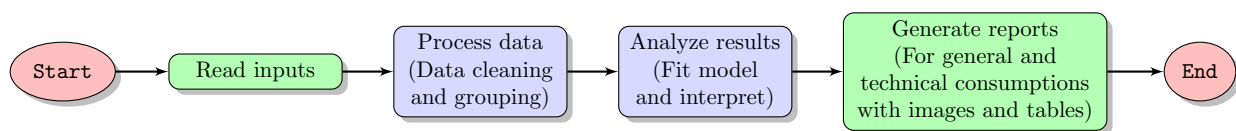


Figure 3: A high-quality flowchart created using the TikZ package.

The TikZ package, based on T_EX is an excellent and elaborate package (manual having > 1300 pages) that can be used for creating high-quality graphics that serve the needs of any technical documentation (Example Fig. 3). Going through the manual of TikZ and the gallery will give information on the package capabilities and how that can be used in the dissertation.

14.3.3 Subfigures

Multiple figures (subfigures) under a common caption can be handled through `subfig` package. The subfigures can be individually sized, captioned, labeled, and referenced. The sub-caption numbering is “alphabetic” by default (holds 26 — and for more subfigures, other options are available) and will be automatically generated. The number of images that occupy a single row can be readily coded with commands, such as `\subfloat{...}`, `\hspace{...}`, and `newline (\\\)`. Refer to the accompanied “NDSU-Thesis-Extended” document for instruction and examples.

14.3.4 Landscape subfigures

Similar to landscape long tables, subfigures that span multiple pages can be enclosed in `landscape` environment block and to produce landscape multiple pages of subfigures with the same figure number. Refer the “NDSU-Thesis-Extended.tex” for example code.

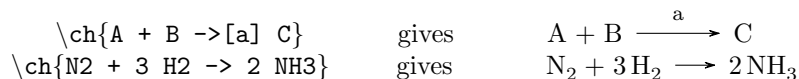
14.3.5 Multiple page subfigures

Sometimes several subfigures running through multiple pages need to be used in the thesis. These are similar to long tables that span several pages. General caption of the set of subfigures fitting in a page are coded with regular `\caption{...}` command, which can be individually controlled. These captions may be the same or abbreviated. Subfigures and their sub-captions were created through `\subfloat[...]{...}` command of `subfig` package. The optional argument of `\subfloat` is the individual subfigure caption and the regular argument is the `\includegraphics` command with its usual optional argument. The `\ContinuedFloat`

with another `figure` environment will carry the numbering forward. When the number of subfigures exceeds the number of alphabets (26), the numbering system should be switched to numeric. Refer the “NDSU-Thesis-Extended.tex” for example code showing multiple page subfigures.

14.4 Chemical symbols

Chemical symbols and chemical equations can be coded easily in a natural manner using the `\ch{...}` command using the `chemformula` package — rather than using the math mode. The following chemicals: H_2O , H_2SO_4 , CrO_4^{2-} , $[\text{AgCl}_2]^-$, $(\text{NH}_4)_2\text{S}$, $^{227}_{90}\text{Th}^+$, and $\text{KCr}(\text{SO}_4)_2 \cdot 12 \text{H}_2\text{O}$ were coded through: `\ch{H2O}`, `\ch{H2SO4}`, `\ch{CrO4^2-}`, `\ch{[AgCl2]-}`, `\ch{(NH4)2S}`, `\ch{^{227}_{90}Th+}`, and `\ch{KCr(SO4)2 * 12 H2O}`, respectively.



Refer to `chemformula` documentation for more options and details. It should be noted that there are other packages available for coding the chemicals and chemical equations, which are not included in the class but users can use them through `\usepackage{...}` command.

14.5 Annotation commands

While developing the dissertation the text undergoes several revisions and suggestions will be provided by the advisor and colleagues. To make suggestions as well as to present the carried out revisions colored annotations will be helpful to draw users’ attention quickly. Therefore, special annotation commands for highlighting, new text, deleted text, replaced text, and notes were defined in the class. These annotation features can be used by the student and the advisor reviewing the dissertation draft. The `ulem` and `todonotes` packages were used to develop these commands, and their documentation may be referred for customization. All the annotations can be searched and deleted before submission, and these processes can be even automated by search expressions (e.g., regular expression). The annotation commands with usage are shown subsequently:

`\hl{Highlight}` gives: **Highlight**. This will be regular text.
`\nt{Test new text.}` gives: *Test new text.* This will be regular text.
`\dt{Deleted text.}` gives: ~~Deleted text.~~ This will be regular text.
`\rt{The text to be deleted}{Which will be replaced by this!}` gives:
~~The text to be deleted~~Which will be replaced by this! This will be regular text again.

While using the above annotation commands, except for `\nt{...}`, enclosing a cited reference commands (`\citep{...}` or `\citet{...}`) use `\mbox{...}` around the cited references. For example,
`\dt{...text...\mbox{\citep{daly2010natural}} ...text...}` gives: ~~...text... (Daly, 2010) ...text...~~

`\notes{To Do notes - for interactive communication!}` (also the shortcut `\td{...}`) gives:

To Do notes - for interactive communication!

14.6 Clever reference — cross-referencing items and labels

Referring to items automatically using the defined labels is a common activity in \LaTeX and is called cross-reference. Although there are basic commands available to refer (e.g., fig. `\ref{label}`), the use of `cleveref` package is an efficient way to achieve this task. This package enhances \LaTeX ’s cross-referencing to automatically detect the “type” of the item cross-referenced (e.g., equation, section, tables, figures, etc.) based on the context of the cross-reference. This means a single command of `\cref{label}` or `\Cref{label}` with the label will produce the correct output (e.g., fig. 1.1, eq. 3, Figure 1.1, Equation 3, etc.). Refer to this package for more details and customization. However, `cleveref` commands will not work with the appendix tables and appendix figures only, where the basic commands (`\ref{...}`) should be used.

14.7 Bibliography compilation issues

The common issues [and proposed solution] while working with bibliographies, especially in combination with `natbib` include:

- The basic principle usually gets violated is that the \LaTeX inputs (`tex`, `bib`, or `others`) require only the ASCII characters. Anything that disregards this principle will have issues during compilation. Presence of extraneous characters (unprintable and formatting) in the `*.bib` file, which comes while copying a formatted non-ASCII text, stalls the compilation [remove them from `*.bib` file]
- Proper title case will not appear in the output (e.g., proper nouns like country names) even though things may seem right in `*.bib` file [use additional curly braces `{...}` to enclose the desired title case items or the letters]
- Proper italics will not show in the output (*Scientific names* of animals and plants; e.g., *Homo sapiens*, *Zea mays*) [use `\emph{...}` around the entries in the `*.bib` file]
- Proper formatting of URLs with wrapping content and clickable links does not appear [use `\url{...}` command in the `*.bib` files entries that have the URLs]
- Outputs not formatted according to the thesis/journal requirement [select the `bst` file that matches most of the requirements and modify the code to suit our requirement might sometimes require — most often only a few alterations are needed and have the file in the same folder as `*.bib`]
- New bibliography styles that are not listed (Sec. 11.1.1) might not run properly, mostly because of not having the corresponding `*.bst` in the system [download style files and have it in the system/local folder]
- Some numerical styles (e.g., IEEEtran family) will not work when the `natbib` was set to author-year format [optional arguments of `natbib` package or arguments of `\citestyle{}` command should be made compatible to the numerical styles (e.g., “numbers” or “plain” in their respective commands) — so a combination of chosen `*.bst` and `citestyle` option is the key to create the references listing in the correct format] `\citestyle egu, agu,`
- While trying different styles, compilation stops even for compatible styles [this is a common phenomenon and could be solved with an understanding of how BibTeX and \LaTeX resolves references by creating several auxiliary files (e.g., `*.aux`, `*.toc`, `*.log`, `*.out`) including `*.bbl` — as contents of these files were referred from unsuccessful compilation, removing all these files leaving the source (`*.tex`) and recompiling with the compatible style and options will restore and regenerate the output. Thus, starting with only a clean source will work and regenerate the necessary files including the output, this strategy will work in other situations of compiling issues as well — however, is not always necessary].

14.8 Proper development of bibliography bib file

- Bib files should not be considered a dumping ground for bibliography codes that were usually not looked at for correctness, unlike the regular document source codes. This sometimes proves to be a serious mistake and stops compilation or produces strange outputs or major errors.
- The bib file, therefore, should be developed step-by-step carefully and each entry should be checked for correctness (only with ASCII characters and allowed equivalent \LaTeX commands for non-ASCII characters) and appropriateness (required and optional fields in bibliography entry).
- Bib files automatically generated by other reference management software (e.g., Zotero) for a given set of references, while they are helpful and provide a good starting point, are sometimes sources of compilation errors. These software may include “abstract” fields as well in the bib file, which are long and become sources of several unintended non-ASCII characters. These superfluous fields may be deleted so that the bib file is compact. Some of these software may include modern entries and fields that may be recognized by Bib \LaTeX but not included in BibTeX. Though it may be possible to create simpler and cleaner bib files, it is a good idea to check the generated bib file “line-by-line” for correctness.
- It was found that “Google Scholar” generated BibTeX entries worked well in developing the bib file. These entries usually generate logical naming of the keys and valid \LaTeX commands for non-ASCII characters

(especially international accented characters). Even in these entries checking the bib entries is always a good idea (e.g., use of capital letters, and emphasis).

- Common errors include:
 - Extraneous non ASCII characters (remember we have demonstrated this during one of our classes);
 - % sign - which is an innocent symbol, but with LaTeX anything after will be considered as a comment so to have % we need to use %;
 - On a similar note, we have a set of reserved symbols (&, %, \$, {, }, , -, #, and so on) that carry a special meaning and should not be used (without preceding) loosely and expect a logical outcome;
 - Every field (author, year, journal, etc.) should end with a comma (but the last one) - in a good bib file, just remove one comma and see the effect - which will teach us how a small thing will wreck the whole;
 - Unconventional fields (though created by other software) should be replaced by the correct ones (Wikipedia BibTeX), for example, date instead of the year will not work;
 - Special symbols like degrees, copyright, and others commonly used in other systems can be coded in LaTeX as regular commands - and that command version of these symbols should be used in a bib or tex file;
 - Duplicate entries.
 - Unmatched braces.
 - Several types of BibTeX entries (<https://en.wikipedia.org/wiki/BibTeX\protect\let\futurelet\@let@token>; e.g., `article`, `book`, `inproceedings`, `masterthesis`, `phdthesis`, `misc`, and so on) with their specific *Required fields* and *Optional fields*. Missing a required field (though generated by other resources) will be a mistake and will suspend compilation. The entry *misc* does not contain any required field and can be used to fit odd entries.
 - The foreign accented characters (e.g., é, ò, ê, ö, ç, ñ, Ì, æ, å, and so on, that are created by `\’e`, `\’o`, `\^e`, `\^o`, `\c c`, `\~n`, `\l`, `\L`, `\ae`, `\aa` L^AT_EX commands) (check the LaTeX cheat sheet and learn how to code them - all start with followed by some ASCII symbols or characters and the input character that receives the accent. As a practice try to have these accented characters in the tex file. If it runs there then that can be inserted into bib.
- While building your bib file add a couple of entries and check them how they compile. Don’t go for the whole 120 entries (just to throw a good number).
- To visualize the reference listing of all the entries issue the command (we have come across this). Later this can be removed and the listing of only cited references will appear - nothing more nothing less.
- Further, remember the use of enclosing capital letters, especially in titles, for example, NDSU, Argentina to get the capitals right from bib files. Similarly the use of *Zee maize* for technical names in bib files for italicized output - otherwise they come out flat.
- It is a good idea while working with bib entries or in the main text, the liberal use of and comment environment for efficient compilation.
- Automatic finding and replacement of non-ASCII

14.9 Temporary ending

While reviewing/revising a large document, it will be efficient to compile only the chapter/section that was currently working. Rather than simply ending the document, it will be helpful to generate the citations in the text and a list of references at the temporary end. As the commands that generate the reference listing need two files (style: `*.bst` and bibliography database: `*.bib`), which will be varying with users, it is better to code this command in the preamble (`\makebib`) as given in Sec. 12.1.2.

For this purpose, a shortcut `\tend` is already defined in the class. Once `\tend` is issued at any stage inside the source code will activate this temporary ending feature and end the document with a reference listing of all the cited references.

14.10 Spacing adjustment around non-textual elements

Usually, the spacing around the non-textual elements produced by L^AT_EX will be good and based on typography principles. The environments that create these elements (e.g., tables, figures, equations) automatically supply an additional space to set the elements apart from the regular text and this is the expected and correct behavior. However, sometimes additional space will appear above or below these elements, which may be the result of fitting the elements with respect to others of the whole chapter. However, the spacing around the non-textual elements can be altered by one or any combination of the following to produce a consistent spacing around the non-textual elements:

- The blank line coded, usually left between paragraphs, might create additional space before the element (e.g., `equation`, `align`) and that can be removed to reduce the space above the element.
- Proper use of vertical spacing `\vspace{...}` command with negative spacing arguments (e.g., `\vspace{-3ex}`) can be able to correct the blank space above the element. This can also be used when a blank line was issued to separate the regular text from the element. Positive vertical space can also be issued as needed.
- When a set of equations was coded (e.g., `align`, `eqnarray`), it will be treated as a block and will not break and flow through multiple pages and gets pushed to the next page. This will create large gaps and can be broken into two or more subsets of equations to fit the page by repeating the environments.
- The actual space around the equations (displayed items) is controlled by the `\abovedisplayskip[=] glue` and `\belowdisplayskip[=] glue`. The glue is called a “rubber” length stating a basic length with an allowed play on both positive and negative sides. The default value for these commands was “12pt plus 3pt minus 9pt”, and is also valid to use the basic length directly as:

`\abovedisplayskip=-12pt`

Another way for issuing the command is using the basic `\setlength` as `\setlength{\abovedisplayskip}{-12pt}`. To have the regular behavior subsequently, the default should be restored by reissuing the commands using the default values.

- In figures, the space above the caption (the space between the bottom of the image and the top of the caption) can be controlled by using the optional argument of the `myfig`, `myfigls`, `myfigap` and `myfigapls` commands. This optional argument was specifically developed to address this caption placement issue. This may be required only for necessary adjustments as the default (without option) will work well in most cases.

14.11 Line numbers for the whole document

Sometimes using line numbers will be helpful while communicating with the advisor or others, where specific locations of the document can be pointed. Line numbers are generated using the package `lineno`, which is coded into the class, by the following command:

`\linenumbers`

This command can be issued at the beginning or at any point, and numbers will appear in the left margin after the command. Of course, this command should be removed or commented while finalizing the thesis.

14.12 Chapter styles

Two styles namely, regular- and paper-styled chapters are generally followed. The regular is a traditional style where the whole thesis/dissertation is considered as a single document where individual chapters exclusively deal with aspects like introduction, literature review, methods, results, discussion or results and discussion, references, and appendices reflecting all studies carried in the research on these individual chapters. Even though this style produces a consolidated document and is solid in its own merit, which ties all research

aspects of the study together in corresponding chapters, a good deal of rewriting will be necessary from the authors if they want to publish the contents as individual peer-reviewed journal articles.

The paper-styled chapters are stand-alone chapters complete with all sections (abstract, introduction, literature review, . . . , references) and are the modern trend. In this style, some amount of repetition among chapters is unavoidable (especially in methods, analysis, and references). However, as the chapters are already in paper-style, it is very easy to format them to suit the requirement of any peer-reviewed journal for submission. It is also possible to have individual chapter references (Bib \LaTeX) or a combined reference chapter (both Bib \TeX and Bib \LaTeX). With \LaTeX it is easy to create stand-alone papers with references for submission from a paper-styled disquisition with a combined reference chapter. As outlined earlier, the commands that start these chapters are `\myheading{...}` or `\mypaperheading{3 args}` (Sec. 8). It is a good idea to consult the advisor before committing to these styles, for they are different and substantial rewriting is involved to switch back and forth.

14.13 Chapters as individual files

When the length of chapters gets long it will be better managed into individual `*.tex` files. Then the thesis file will become a collection of such individual files and will be highly compact. The individual chapters are coded using either of these commands:

```
\input{filename}
\include{filename}
```

The `\input{filename}` imports the codes from the `filename.tex` into the main file at the location where this command was issued. This is equivalent to typing all the code commands from the individual file into the main file. The `\include{filename}` issues a `\clearpage` before and after inserting the contents and had better speed than the `\input{filename}` command. With such commands in place, it is possible to compile only the chapter the user wants to work on by commenting others, and this approach saves unnecessary compilation or reduces compilation times.

14.14 Chapters as subfiles

Another useful method of handling chapters is the application of subfiles (Sec. 14.17) that allows for individual files to be compiled and generate outputs. Subfiles are standalone documents that derive the preamble from the main document (e.g., `my-dissertation.tex`) and represent each chapter (e.g., `chapter1-intro.tex`, `chapter2-methods.tex`). The text of the “subfile” will have the following structure:

```
\documentclass[my-dissertation.tex]{subfiles}
\begin{document}
... Entire chapter’s text from title, abstract, sections/subsections, and
   so on, with citations using citation and bibliography commands ...
\end{document}
```

While the “main” file (`my-dissertation.tex`) will have the following structure (similar to Sec. 3 “Thesis Example”):

```
\documentclass[options]{ndsu-thesis-2022}
... Preamble information ...
\begin{document}
\subfile{chapter1-intro.tex}
\subfile{chapter2-methods.tex}
... Bibliography commands, if required ...
\end{document}
```

It can be observed that the main file assembles all the chapters in subfiles, while each subfile borrows the class and preamble from the main file. All the subfiles can be kept in the same folder of the main file or can be stored separately in a subfolder but appropriately adding the paths in both the files (Refer to the additional example in the class bundle for the use of subfiles and paths).

14.15 Defining and using specific commands, environments, and packages

As it is not possible to write a class to satisfy the specific requirements of the several departments of NDSU, most of the major features as outlined in this document were coded into the NDSU class, and the users can add necessary features in their source code specific to their requirement. This approach gives flexibility making the class compact and useful. Mathematics, physics, chemistry, literature, etc. departments may use specific symbols and environments that other departments (e.g., agriculture) do not for their thesis. This means that based on the requirement, if not already the packages already loaded in the class, the necessary packages can be loaded in the source files and they work with the class. Another aspect is bibliography style and management, which varies with different departments. To deal with this the necessary style files (*.bst) should be kept in the same folder as *.tex or the appropriate path specified in the source code.

Specific packages (e.g., bibliography management), new commands (shortcuts), and new environments (e.g., theorem, proof, etc.) can be included in the preamble. These specific items are best left to individual users, as others may not need this — hence they are not included in the class deliberately.

14.16 Thesis to journal article conversion for submission

A few adjustments are necessary to convert the thesis chapters into journal articles. The commands that are specific to the class (e.g., `\myfig{}`, etc.) will not be known to the journal L^AT_EX template. When the corresponding `\newcommand` codes (made of basic commands) as shown in Table 2 against the class shortcut below were inserted in the template preamble it will compile. As the shortcuts are convenient and reduce the amount of code typed (Don't repeat yourself — DRY method), it will be better to insert the preamble and retain the thesis code than to replace it with basic commands at all occurrences.

Table 2: Converting thesis chapters to journal articles — preamble information to be added to journal template source code.

Shortcut example commands in thesis (Refer Sec. 10.2.1)	Command to be inserted in the preamble for journal articles
<pre>\myfig[0.4cm] {h!}{1.0}{fig1.png} {Regular figure caption} {fig:1}</pre>	<pre>\newcommand{\myfig}[6][0ex]{ \begin{figure}[#2] \centering \captionsetup{aboveskip=#1 plus 2pt minus 2pt, belowskip=0pt} \includegraphics[width=#3\textwidth]{#4} \caption{#5} \label{#6} \end{figure} }</pre>
<pre>\myfigls[0.5cm] {ht}{0.8}{fig2.pdf} {Landscape figure caption} {figls:2}</pre>	<pre>\newcommand{\myfigls}[6][0ex]{ \begin{landscape} \begin{figure}[#2] \centering \captionsetup{aboveskip=#1 plus 2pt minus 2pt, belowskip=0pt} \includegraphics[width=#3\textwidth]{#4} \caption{#5} \label{#6} \end{figure} \end{landscape} }</pre>

The compatibility issue will not be there with the tables for no shortcuts were defined in the class and all table codes used the basic commands from the respective packages. Also, the heading commands (`\myheading{}` and `\mypaperheading{}`) are specific to the thesis class (See Sec. 8) and will not generally work with journal template. However, their different arguments can be extracted and used in the commands specific to the journal template. Similarly the code from the appendices such as, `appendixable` and

`appendixfigure` environments should be replaced by the basic `table` and `figure` environments, respectively, in the article template. Needless to mention that the NDSU class-specific front matter commands that may not appear in individual chapters such as, `author`, `date`, `progdeptchoice`, `department`, `cchair`, `cmember(a-d)`, `approvaldate`, `approver`, `listofabbreviations`, `listofsymbols`, and so on, will not work in the journal template; but the content of the arguments can be copied and utilized in the appropriate journal template commands.

It may be also necessary to include the missing packages in the template, which were used in the thesis class. The missing packages can be identified from the “Undefined control sequence” error messages from the command that stop the compilation (e.g., using `\cref{}` without loading the `cleveref` package). Of course, the template-specific commands should be used while developing the journal articles. With these adjustments, the thesis code easily transitions to the journal articles. It is hoped that with the working knowledge of L^AT_EX previously obtained and/or gathered thus far the students can able to handle what is required to convert the thesis chapters into journal articles source code utilizing the specific journal templates.

14.17 Advanced options

move to additional items

Although not required for simple work, a few class options were made available to further automate the dissertation writing. The first option is `[chaptersbib]`. This is intended to produce individual chapter bibliography and should only be used with BibL^AT_EX and BibT_EX will not support chapters bibliography — based on this class. This option works with chapters processed as “subfiles,” where those chapters can be individually compiled and output generated.

The second option is `[subfileref]`. Subfiles, usually individual chapters, can be processed individually following the same style of the whole document. The information in the preamble of the main dissertation document (that pulls in and assembles all the chapters of the dissertation in individual files) was again derived and used as the preamble of the subfile. The required package “`subfiles`” was included in the class. For each chapter to be standalone, like a paper, the chapter should have the relevant bibliography as the last unnumbered section/chapter, and using this option in the main document will accomplish this. This option work with both BibL^AT_EX and BibT_EX . The whole dissertation processing, obtaining contents from these individual subfiles, with intended combined bibliography chapter, should not also produce the individual chapters bibliography, therefore this option should not be used. Refer to the additional example in the class bundle that shows the necessary codes in the main and chapter files.

The following table shows the various outcomes given the options (Table 3):

Table 3: Options for degree and disquisition types

Option	Outcomes
<code>[chaptersbib]</code> only	Dissertation with individual chapters bibliography. Works with BibL ^A T _E X only.
<code>[subfileref]</code> only	For individual chapter processing bibliography. Works with both BibL ^A T _E X and BibT _E X.
<code>[]</code> none — default	Dissertation with combined bibliography as unnumbered chapter.
<code>[chaptersbib,subfileref]</code> both	<i>Don't use.</i> Possible redundant bibliographies. Requires BibL ^A T _E X.

14.18 Limitations of the class

- *Shortcuts for tables:* Unlike figure environment, table vary a lot (e.g., number of columns and rows) hence not possible to develop shortcuts (such as `\myfig{}`). Therefore, no shortcuts were coded in the class and the users should go for the basic command using the `\table` environment.
- *Citations in captions:* This means using `\citep{...}` or `\citet{...}` or `\cite{...}` inside the `\caption{...}` argument of tables and figures is not supported. However, the output of the cites can

be hard-coded (after noting the output outside of the caption) as a workaround. It should be noted that cite commands are known to work in captions of other classes.

- *Limited packages loaded:* Only limited packages that are thought to be useful for students and cater to most thesis/dissertations, in general, are included in the class. Therefore for special needs, users need to include the respective packages in the preamble.

15 Additional Information II — Some Tips For Customization

15.1 General suggestions

NDSU Graduate School Formatting Guidelines (<https://www.ndsu.edu/sites/default/files/2021-09/Format-Guidelines-2021.pdf>) should be adhered to and the guidelines to be referred for various aspects of developing the work. Following are some of the general suggestions while developing the thesis/dissertation:

- All content-related decisions should be made by the student, advisor, and committee, and should follow any rules or conventions established within your program, department, or field.
- Students are highly encouraged to follow the prevalent style manual of the discipline (e.g., MLA, APA, Chicago, IEEE, etc.) while formatting the thesis (especially formatting of tables, figures, and other non-textual elements). In the instances where the Graduate School guidelines contradict the style manual for your discipline, the former takes precedence. However, if a generic style is applied consistently throughout all items it will also be approved. During format review, a consistent application of one style is accepted. In short, consistency is KEY.
- Paper-based or regular thesis/dissertation chapters (two possible styles) should follow NDSU format guidelines consistently across all chapters and use the prevalent style manual of the discipline.
- References, tables, and figures should follow the most appropriate style manual of the discipline. Some have the caption centered and set in bold font.
- A footnote should be included if the chapter is co-authored (an example of this is in NDSU guidelines) and including publication information in the footnote or in the Acknowledgments section is recommended.
- The general recommendation about spacing between items and their surrounding non-textual content (equations, figures, tables, quotes, pseudocode, etc.) is to set a consistent spacing between items and their surrounding content as seen in most academic publications.
- Table (and figure) captions should be in the same font size as general text; however, text inside tables and footnotes may be in smaller font sizes as needed to fit the item within the page margins.
- NDSU guidelines have a number of very specific rules (e.g., Table of Contents and Prefatory List formatting, abstract word count, headings, body text paragraphs, etc.); however, they give a lot of flexibility for what is not covered, and give the student (and committee) control of the written content.
- It should be noted that L^AT_EX is a vast program with numerous facilities and resources that students can use while developing their thesis/dissertation and improving the document quality. All regular L^AT_EX commands and features work well with the NDSU thesis class.

15.2 Voice of the T_EX and L^AT_EX developers!

It is interesting to know what the original developers of T_EX and L^AT_EX have to say about this system of document preparation. Following are the quotes from the developers about how people feel, perceive, and use the system for their documentation needs.

“I never expected T_EX to be the universal thing that people would turn to for the quick-and-dirty stuff. I always thought of it as something that you turned to if you cared enough to send the very best.”
— Donald Knuth (Developer of T_EX [on which L^AT_EX is based])

“ \LaTeX is easy to use — if you’re one of the 2 % of the population who thinks logically and can read an instruction manual. The other 98 % of the population would find it very hard or impossible to use.”

— Leslie Lamport (Developer of \LaTeX)

It is safe to assume that students who came this far should have “cared enough” to improve the quality of their thesis/dissertation, and some who may think they are in the 98 % might discover that they have better logical skills than they originally believed. Furthermore, using \LaTeX for the documentation needs (e.g., thesis/dissertation, paper, report, book, letter, CV, and so on) should be considered a useful skill in itself that students can pick up and use throughout their carrier.

Acknowledgements

The authors gratefully appreciate the leadership and inputs from Danjel Nygard, Dissertation and Thesis Coordinator, NDSU, Fargo, ND, in developing this “ndsu-thesis-2022” \LaTeX class. The support and involvement of James Thorne, Ph.D. Student, Department of Mathematics, NDSU, Fargo, ND, the maintainer of the previous version of this class in CTAN, is also highly appreciated.

Index

abovedisplayskip, 23	whole, 14	phd, 7
abstract, 10	BibTeX, 14, 15, 26	deleted text, 20
acknowledgements, 10, 28		department, 9
align, 23	caption, 13	disquisition approval, 9
annotations, 20	captionsetup, 13	document, 11
appendix, 17	cchair, 9	document frames, 7
appendixfigure, 17	chapter, 11	documentclass, 6, 7, 16, 24
appendixtable, 17	chapter styles, 23	Donald Knuth, 27
LOAF, 17	chaptersnumber, 7	doublespacing, 18
LOAT, 17	chaptersbib, 26	draft, 7
myfigap, 17	chemical formula, 20	DRY, 25
myfigapls, 17	chemical symbols, 20	dummy images, 12
author, 9	citep, 15	dummy text, 12
automatic components, 10	citet, 15	
	class, 2	eqnarray, 23
belowdisplayskip, 23	cleveref, 20	equation, 13
Bib \LaTeX , 15	closeappendices, 11, 17	example images, 12
addbibresource, 15	closeappendixfigures, 11,	example thesis, 2
biber, 15	17	
multiple	closeappendixtables, 11, 17	figure, 10
bibliographies,	co-authored, 27	caption, 13
15	cochairZ, 9	centering, 13
options, 15	columnwidth, 18	includegraphics, 13
printbibliography, 15	compilation times, 24	label, 13
bibliography, 14	Cref, 20	landscape, 13
BibLaTeX, 14	cref, 20	landscape subfigures,
BibTeX, 14	cross-referencing, 20	19
citep, 15		myfig, 13
citestyle, 14	date, 9	myfigls, 13
citet, 15	dedication, 10	placement, 13
compilation, 21	default behavior, 6	resize, 13
individual chapter, 14	degree, 9	subfigures multipage,
issues, 21	ma-paper, 7	19
natbib, 14	ma-thesis, 7	flowchart, 19
stand-alone chapter,	ms-paper, 7	font, 8
14	ms-thesis, 7	bookman, 8

<ul style="list-style-type: none"> charter, 8 gentium, 8 kpfonts, 8 libertine, 8 mathdesign, 8 mathptmx, 8 newcent, 8 palatino, 8 tgbonum, 8 tgpagella, 8 tgschola, 8 tgtermes, 8 times, 8 utopia, 8 zlmitt, 8 	<ul style="list-style-type: none"> linenumbers, 23 list of abbreviations, 10 list of appendix figures, 11 list of appendix tables, 11 list of figures, 10 list of symbols, 11 list of tables, 10 LOA, 10 LOAF, 11 LOAT, 11 LOF, 10 LOS, 11 LOT, 10 	<ul style="list-style-type: none"> spacing, 23 specific commands, 25 specific packages, 25 standalone editor, 2 style manual, 27 subfigures, 19 subfileref, 26 subfiles, 24, 26 subfloat, 19
<ul style="list-style-type: none"> font options, 8 font sizes, 7 fully-justified, 7 	<ul style="list-style-type: none"> makebib, 15 maketitle, 11 margin frame, 7 margin overflows, 7 mwe, 12 myspacing, 18 	<ul style="list-style-type: none"> table, 10 <ul style="list-style-type: none"> arraystretch, 18 booktabs, 12 caption, 12 columnwidth, 12 footnote, 12 full-width, 12 landscape, 12, 18 landscape longtable, 18 longtable, 12 pdflscape, 12 resizebox, 12 tabcolsep, 13 tablenotes, 18 tabular, 12 tblr, 12 threeparttable, 12
<ul style="list-style-type: none"> general suggestions, 27 geometry, 7 graphicspath, 19 	<ul style="list-style-type: none"> namedappendices, 17 natbib, 8, 14 natbib styles, 14 negative spacing, 23 new commands, 25 new text, 20 nojustify, 7 nonumber, 7 	<ul style="list-style-type: none"> table of contents, 10 template, 2 temporary end, 22 tend, 22 theorem environment, 8 thesis to paper, 25 tikz, 19 tikzpicture, 19 title, 9 TOC, 10 todonotes, 20
<ul style="list-style-type: none"> heading, 11 <ul style="list-style-type: none"> myheading, 11 mypaperheading, 11 paragraph, 11 section, 11 subsection, 11 subsubsection, 11 highlighting, 20 	<ul style="list-style-type: none"> online editor, 2 options, 6 overflow, 7 	<ul style="list-style-type: none"> vertical spacing, 13 vspace, 23
<ul style="list-style-type: none"> include, 24 individual files, 24 input, 24 	<ul style="list-style-type: none"> placeholder, 7 preface, 10 program, 9 	
<ul style="list-style-type: none"> journal article, 25 	<ul style="list-style-type: none"> reference listing, 8 refsection, 16 replaced text, 20 	
<ul style="list-style-type: none"> kantlipsum, 12 	<ul style="list-style-type: none"> showframe, 7 	
<ul style="list-style-type: none"> left-justified, 7 Leslie Lamport, 28 limitations of class, 26 		

— ★ ★ ★ —