

The hagenberg-thesis Package

W. Burger and W. Hochleitner

University of Applied Sciences Upper Austria
Department of Digital Media, Hagenberg (Austria)

2021/05/12

Abstract

The `hagenberg-thesis` package is a collection of modern LaTeX templates for university theses (bachelor, master or diploma programs) and related documents. This manual describes the main features of this package. Pre-configured document templates for English and German manuscripts and a complete tutorial are available on the package's home repository.

1 Introduction

The complete source of this package and auxiliary materials are available on CTAN¹ and its development repository.² The package is made available under the terms of the Creative Commons Attribution 4.0 International Public License.³

2 Document classes

The `hgb` package provides the following document classes, which are based on the standard LaTeX classes `book`, `report` and `article`, respectively:

- `hgbthesis` (`book`): for bachelor's, master's and diploma theses;
- `hgbreport` (`report`): for project and term reports;
- `hgbarticle` (`article`): for drafting journal articles.

2.1 Class options

All document classes accept the following **general options**:

- `english` or `german` (select the primary language),
- `smartquotes` (use smart quotes replacement),

¹<https://ctan.org/pkg/hagenberg-thesis>

²<https://github.com/Digital-Media/HagenbergThesis>

³<https://creativecommons.org/licenses/by/4.0/legalcode>

- `apa` (use `apa` bibliography style instead of `numeric-comp`),
- `noUpdateCheck` (suppress check of package release date).

In addition, the following **class-specific options** are accepted:

- `hgbthesis: master, diploma, bachelor, internship`;
- `hgbreport: notitlepage`;
- `hgbarticle: twocolumn`.

For example, to start a master's thesis in German one would simply place

```
\documentclass[master,german,smartquotes]{hgbthesis}
```

at the beginning of the document.

2.2 Thesis parameters (class `hgbthesis`)

`hgbthesis` supports several types of thesis documents. The following parameters must be specified for *all* types:

- `\title{...}`,
- `\author{...}`,
- `\programtype{...}`,
- `\programname{...}`,
- `\placeofstudy{...}`,
- `\dateofsubmission{yyyy}{mm}{dd}`,
- `\advisor{...}` (optional).

Note that `hgbthesis` only supports a *single author* inside the `\author{...}` macro argument (commands `\and` and `\thanks{...}` are deactivated)!

3 Style files and user commands

The package comes with a set of style (`*.sty`) files that can be used independently of the document classes listed above: `hgb.sty`, `hgbabbrev.sty`, `hgbbib.sty`, `hgbheadings.sty`, `hgblistings.sty`, `hgbmath.sty`.

3.1 General user commands and environments(`hgb.sty`)

- `\hgbDate`: Outputs the package version date, e.g., “2021/05/12”.
- `\calibrationbox{width}{height}`: Inserts a test box for checking the final print size (in millimeters).
- `\begin{english} ... \end{english}`
- `\begin{german} ... \end{german}`

3.2 Text commands (`hgbabbrev.sty`)

Special characters:

- `\bs`: Inserts a backslash character (short for `\textbackslash`).
- `\obnh`: Inserts an optional break with no hyphen (e.g., `PlugIn{\obnh}Filter`).

German abbreviations:

- `\bzgl:` bzgl.
- `\bzw:` bzw.
- `\ca:` ca.
- `\dah:` d. h.
- `\Dah:` D. h.
- `\ds:` d. sind
- `\etc:` etc.
- `\evtl:` evtl.
- `\ia:` i. Allg.
- `\sa:` s. auch
- `\so:` s. oben
- `\su:` s. unten
- `\ua:` u. a.
- `\Ua:` U. a.
- `\uae:` u. Ä.
- `\usw:` usw.
- `\uva:` u. v. a.
- `\uvm:` u. v. m.
- `\va:` vor allem
- `\vgl:` vgl.
- `\zB:` z. B.
- `\ZB:` Zum Beispiel

English abbreviations:

- `\ie:` i.e.
- `\eg:` e.g.
- `\etc:` etc.
- `\Eg:` E.g.
- `\wrt:` w.r.t.

Note that none of the above abbreviation macros “eats” subsequent white space, i.e., they can be used without additional controls, as in “`\wrt what I said`”, for example.

3.3 Bibliography commands (hgbbib.sty)

- `\AddBibFile:` A wrapper to biblatex’s `\addbibresource` macro (for backward compatibility only).
- `\MakeBibliography[options]:` Inserts the reference section or chapter. By default, references are automatically split into category subsections.⁴ Use the option `nosplit` to produce a traditional (i.e., contiguous) list of references.

⁴Predefined reference categories are `literature`, `avmedia`, `online` and `software`.

- `\citenobr{keys}`: Analogous to the standard `\cite{keys}` command, but inserts no “backref” page numbers in the bibliography.
- `\mcite[text1]{key1}[text2]{key2}...[textN]{keyN}`: Analogous to `biblatex`’s `\cites` command,⁵ but inserts semicolons between reference entries for better readability.

3.4 Code environments (`hgblistings.sty`)

The following types of code environments are defined:

- `CCode`: for C (ANSI),
- `CppCode`: for C++ (ISO),
- `CsCode`: for C#,
- `CssCode`: for CSS,
- `GenericCode`: for generic code,
- `HtmlCode`: for HTML,
- `JavaCode`: for Java,
- `JsCode`: for JavaScript,
- `LaTeXCode`: for LaTeX,
- `ObjCCode`: for ObjectiveC,
- `PhpCode`: for PHP,
- `PythonCode`: for Python,
- `Swift`: for Swift,
- `XmlCode`: for XML.

`hgblistings` is based on the `listingsutf8`⁶ package, thus any valid `listings`⁷ option may be used; for example, the option `numbers=none` to suppress line numbers:

```
\begin{JavaCode}[numbers=none]
... // Java code comes here
\end{JavaCode}
```

3.5 Mathematical commands (`hgmath.sty`)

`hgmath` requires (and automatically loads) the `amsmath`⁸ package, thus all commands and symbols of `amsmath` are available by default. The following *additional* commands can only be used in math mode:

- `\Cpx`: \mathbb{C} (complex numbers),
- `\N`: \mathbb{N} (natural numbers),
- `\R`: \mathbb{R} (real numbers),
- `\Q`: \mathbb{Q} (rational numbers),
- `\Z`: \mathbb{Z} (integer numbers).

⁵<http://mirrors.ctan.org/macros/latex/contrib/biblatex/doc/biblatex.pdf> (see Sec. 3.8.3)

⁶<https://ctan.org/pkg/listingsutf8>

⁷<https://ctan.org/pkg/listings>

⁸<https://ctan.org/pkg/amsmath>

3.6 Algorithms (hgbalgo.sty)

`hgbalgo` is a stand-alone package that is based on – and extends – the `algorithmicx` and `algpseudocodex` packages.⁹ It fixes some (mostly indentation-related) problems, adds color and provides some additional commands. It also loads the `algorithm`¹⁰ package which defines a compatible float container for algorithms: `\begin{algorithm} ... \end{algorithm}`.

Additional user commands:

- `\StateNN[<nesting>]{<text>}`: Creates a *non-numbered* statement like `algorithmicx`'s `\State` command but provides controlled indentation inside nested constructs. The optional integer argument `<nesting>` can be used to specify the *nesting depth* to compensate for a bug in `algorithmicx` (the nesting level inside a block is not set properly before the first `\State` command). Omitting the optional argument should give correct indentation in most cases.
- `\Input{<text>}`: For describing the input parameters in a procedure's preamble.
- `\Output{<text>}`: For describing the output values in a procedure's preamble.
- `\Returns{<text>}`: For describing the return values in a procedure's preamble.

Vertical spacing commands: The following commands are provided for fine-tuning the vertical spacing between individual statements of an algorithm (the standard spacing commands like `\smallskip` etc. have no effect between statements):¹¹

- `\algsmallskip`: inserts 3pt extra space,
- `\algmedskip`: inserts 6pt extra space,
- `\algbigskip`: inserts 12pt extra space.

They are supposed to be used inside (i.e., at the end of) statements, for example:

```
\State $x \gets x + 1$ \algsmallskip
```

Defined algorithm colors:

- `AlgKeywordColor` (for algorithm keywords),
- `AlgProcedureColor` (for procedure and function names).

These colors can be redefined at any time (see the `xcolor`¹² package), e.g., by

```
\definecolor{AlgKeywordColor}{named}{black}
\definecolor{AlgProcedureColor}{rgb}{0.0, 0.5, 0.0} % dark green
```

⁹<https://ctan.org/pkg/algorithmicx>, <https://ctan.org/pkg/algpseudocodex>

¹⁰<https://ctan.org/pkg/algorithms>

¹¹Note that the standard spacing commands work *between* procedure and function blocks in the usual way.

¹²<https://ctan.org/pkg/xcolor>

3.7 Abbreviations, Acronyms and Nomenclature (`hgbacro.sty`)

`hgbacro` is a wrapper for the `acro`¹³ package. It provides a simple and pre-configured setup to typeset acronyms or nomenclature.

Acronyms should be defined in a file called `acronyms.tex` in the root directory. By using the `\ac{...}` command, acronyms can be referred in the running text. The extended description is used for the first appearance, whereas the acronym is printed for any subsequent use.

To display a list of acronyms, use the `\PrintAcronyms` command.

For additional options, please refer to the `acro` manual.

4 Package dependencies

The `hagenberg-thesis` package builds on the following LaTeX packages:

`abstract`, `acro`, `algorithm`, `algorithmicx`, `algpseudocodex`, `amsbsy`, `amsfonts`, `amsmath`, `amssymb`, `babel`, `biblatex`, `breakurl`, `caption`, `cmap`, `csquotes`, `datetime2`, `enumitem`, `epstopdf`, `eurosym`, `exscale`, `fancyhdr`, `float`, `fontenc`, `geometry`, `graphicx`, `hypcap`, `hyperref`, `ifpdf`, `inputenc`, `listingsutf8`, `lmodern`, `moreverb`, `overpic`, `pdfpages`, `pict2e`, `subdepth`, `titlesec`, `titling`, `tocbasic`, `url`, `upquote`, `verbatim`, `xcolor`, `xifthen`, `xstring`, `xspace`.

¹³<https://ctan.org/pkg/acro>