

HsH-Classes — A set of \LaTeX classes for use in Hochschule Hannover *

Jan Wille[†]

Created 15.06.2024

Maintained on https://lab.it.hs-hannover.de/qxx-tul-u1/latex-template-hsh

Contents

1 The different classes	2	5.2.2 Default options	10
2 Document options	2	5.2.3 Loading the parent class	11
2.1 Generic options share by all classes	2	5.3 Package loading	11
2.2 Options for modifying the document	2	5.3.1 Ensuring German works	11
2.3 Standalone specific options	3	5.3.2 Generally usefull packages	11
3 Provided commands	3	5.3.3 Options for packages that could be loaded by the user	12
3.1 Title matters	4	5.4 Custom commands	13
3.2 Commands for document writing	4	5.4.1 Document seperation commands	13
3.3 Default \LaTeX Commands that are modified	4	5.4.2 The Logo for Hochschule Hannover	13
3.4 Micalanious commands	4	5.4.3 Title matters	14
4 Package laoding	5	5.4.4 Commands for document writing	15
4.1 Allways loaded packages	5	5.4.5 Micalanious commands	16
4.2 Conditionally loaded Packages	5	5.5 Document setup	16
4.3 Pre-configured Packages	5	5.5.1 Fonts and text styling	16
5 Implementation	5	5.5.2 Page layout	16
5.1 Internal commands	5	5.5.3 Styling \LaTeX default constucts	16
5.2 Option handling	6	5.5.4 Header and footer	17
5.2.1 Unknown options	10	5.5.5 Titlepage	18
		5.6 Localisation	23

Abstract

The following documents a set of \LaTeX classes created for the Hochschule Hannover. They are intended to ease the workflow when writing documents by providing a common formating basis that should work for pretty much everything a studend will be expected to write. This can be simple one-paged documents, excercises, lab-reports, papers or bachelors and masters thesises.

The classes provide interfaces to modify commend requiriements, provide commands to get specifics like the logo and provide and pre-configure comonly needed packages. This should get you going imidealty and reduce the setuptime significantly.

*This file describes version 3.03.pre0, last revised 2024-06-14.

[†]E-mail: mail@janwille.de

1 The different classes

The project classes provided by the Project all carry the HsH- prefix. Here is a list of the available classes and some explanation on when to use which class.

HsH-article	A article-class based on KOMA-Scripts scrartcl. It is designed for quick and compact documents and is useful for writing lab-protocols and alike. It does not have chapters and therefore never breaks to a new page on its own.
HsH-report	A report-class based on KOMA-Scripts scrreprt. This is probably the most useful class, as it can be used for a wide variety of documents (beginning with lab-reports and ending at complete thesis). The line between article and report is somewhat blurry, so use as you see fit.
HsH-standalone	A helper class based based on the standalone class. It is designed only for creating images as separate documents to keep things organized and compiler times low. It is useful for creating graphs, circuit diagrams or other kind of complex sub documents.

2 Document options

To configure the behavior and style of documents using this class, options can be passed via the `\documentclass[options]{document-class}` command.

It should be noted that all unknown keys will be passed to the parent class and a log-message issued.

2.1 Generic options share by all classes

These Options are available regardless of documentclass and modify common things.

fontfamily The `fontfamily=opt` option configures which font-style is used. For convenience there are also short-forms provided. The available options are:

`sans`
`roman`

`sans` | `sans-serif` A sans-serif font is used (similar to Arial)

`roman` | `serif` A serif font is used (similar to Times-new-Roman)

language The `language=opt` option set the main language you write in. It ensures texts like auto-generated headings are localised properly. You can pass in any language-name understood by the babel package. German is the default. For convenience there are also short-forms provided.

`english`
`german`

todos The todos option is a simple switch that activates support for the todonotes package. It enables/disables the package as well as increasing the pagewidth to make space for the notes. You can use the commands from the package to make notes and other annotations (similar to how MS-Words comments work). When you pass `off` | `false` all the annotation will disappear from the PDF while still being in the source.

Note: setting `todos=off` after having used the option will most likely produce compilation errors. These will go away after you remove the temporary files.

2.2 Options for modifying the document

The following options are only available for documents (so not utility classes).

linespacing The `linespacing=opt` option configures the spacing in between lines. For convenience there are also short-forms provided. The available options are:

`singlespacing`
`onehalfspacing`
`doublespacing`

`single` No additional space is added in between lines.

`onehalf` Approximately half a line of empty space is added in between lines.

`double` About a full line height is left in between lines.

parskip The `parskip=opt` option configures the spacing in between paragraphs. This is an extended option originally implemented by KOMA-Script.

<code>never</code>	No inter-paragraph spacing will be inserted even if additional vertical spacing is needed for vertical adjustment with <code>\flushbottom</code> .										
<code>never+</code>	No inter-paragraph spacing will be inserted. There must be at least a third of a line of free space at the end of a paragraph.										
<code>never*</code>	No inter-paragraph spacing will be inserted. There must be at least a quarter of a line of free space at the end of a paragraph.										
<code>...</code>	see KOMA-Script manual, Table 3.7 for more options.										
<code>headheight</code>	The <code>headheight=<dim></code> option allows you to set the required size of the header. You may need to modify this if you get a <code>\headheight to low</code> error message. The message should tell you what value you need, but you can pass any valid length.										
<code>abstract</code>	The <code>abstract=<opt></code> option allows you to configure different behaviors of the abstract. The available options are described below: <table> <tr> <td><code>keywords</code></td><td>Print the <code>\keywords</code> after the abstract.</td></tr> <tr> <td><code>nokeywords</code></td><td>Do <i>not</i> print the <code>\keywords</code> after the abstract.</td></tr> <tr> <td><code>totoc</code></td><td>The abstract will be listed in the table of contents.</td></tr> <tr> <td><code>notoc</code></td><td>The abstract will <i>not</i> be listed in the table of contents.</td></tr> </table>	<code>keywords</code>	Print the <code>\keywords</code> after the abstract.	<code>nokeywords</code>	Do <i>not</i> print the <code>\keywords</code> after the abstract.	<code>totoc</code>	The abstract will be listed in the table of contents.	<code>notoc</code>	The abstract will <i>not</i> be listed in the table of contents.		
<code>keywords</code>	Print the <code>\keywords</code> after the abstract.										
<code>nokeywords</code>	Do <i>not</i> print the <code>\keywords</code> after the abstract.										
<code>totoc</code>	The abstract will be listed in the table of contents.										
<code>notoc</code>	The abstract will <i>not</i> be listed in the table of contents.										
<code>toc</code>	The <code>toc=<opt></code> option configures what is listed in the table of contents. <table> <tr> <td><code>totoc</code></td><td>The table of contents will list itself.</td></tr> <tr> <td><code>notoc</code></td><td>The table of contents will <i>not</i> list itself.</td></tr> <tr> <td><code>abstract</code></td><td>The abstract will be listed in the table of contents.</td></tr> <tr> <td><code>noabstract</code></td><td>The abstract will <i>not</i> be listed in the table of contents.</td></tr> <tr> <td><code>...</code></td><td>see KOMA-Script manual, Table 3.5 for more options</td></tr> </table>	<code>totoc</code>	The table of contents will list itself.	<code>notoc</code>	The table of contents will <i>not</i> list itself.	<code>abstract</code>	The abstract will be listed in the table of contents.	<code>noabstract</code>	The abstract will <i>not</i> be listed in the table of contents.	<code>...</code>	see KOMA-Script manual, Table 3.5 for more options
<code>totoc</code>	The table of contents will list itself.										
<code>notoc</code>	The table of contents will <i>not</i> list itself.										
<code>abstract</code>	The abstract will be listed in the table of contents.										
<code>noabstract</code>	The abstract will <i>not</i> be listed in the table of contents.										
<code>...</code>	see KOMA-Script manual, Table 3.5 for more options										

Often relevant KOMA-Script options

The following options are implemented by the parent classes and only listed here for completeness. For more details, see [its manual](#).

<code>fontsize</code>	The <code>fontsize=<size></code> options takes a size in pt. It is usually in the range of 10-12, but other sizes can work as well.
<code>paper</code>	The <code>paper=<size></code> options accepts a number of options, most ISO formats are supported, but also others like <code>letter</code> or <code>legal</code> .
<code>twoside</code>	The <code>twoside</code> option sets your document up for double-sided printing. The header and footer will take this into account and binding-correction will be applied along the inner edge.
<code>BCOR</code>	The <code>BCOR=<dim></code> option allows you to define a custom binding-correction. Any valid length can be put here, but too large of a value will shrink the outer margin to a not-desirable level.

2.3 Standalone specific options

The standalone utility class has some special options which are documented here.

<code>margin</code>	The <code>margin=<dim></code> option controls how much whitespace is added around your standalone document. This usually looks better which is why the default is 0.25 cm, but you can suppress it by passing 0 cm.
<code>multi</code>	The <code>multi=<opt></code> option defines which environment makes up a page. It can be passed more than once.

3 Provided commands

The classes define a set of commands which are explained in the following section.

<code>\HsHClassName</code>	Each class defines the macro <code>\HsHClassName</code> to contain its class name. This is mostly useful so internals can reuse the class name, but you could also check against it if you needed to.
----------------------------	---

3.1 Title matters

L^AT_EX has a set of default commands which are used to define data for the titlepage, like `\title` or `\author`. The classes define a few additional commands, which are documented here.

- `\matrikelnr` The `\matrikelnr{<nr[, ..>}` macro sets the matrikelnumber of the author(s). It can be a single number or a comma seperated list of numbers. The numbers will be matched to the authors passed into `\author`.
- `\professor` You can pass any text to `\professor{<text>}`, it will be printed on the bottom of the titelpage.
- `\keywords` This macro can be used to define keywords which are relevant to your document. They will be printed as part of the abstract and be put into the PDF's meta-data.

Modifying the Logo

- `\HsHlogoPath` The logo is loaded from a file whos name is read from `\HsHlogoPath`. Change it to use a different logo-file.
- `\HsHlogoPage` As the PDF file can hae multiple pages, the `\HsHlogoPage` command stores which page to load.
- `\includeHsHlogohere` The macro `\includeHsHlogohere[<width>]` is used by `\maketitle` to produce the logo. But if you want it elsewhere, you cann call this macro yourself.

3.2 Commands for document writing

The following commands will be usefull to you when writing a document.

- `\declarationofauthorship` The command `\declarationofauthorship[<align>]` can be used to print a "declatation of authorship" in the current location, similar to how `tableofcontents` and `friends` work. It will produce a horizontal line, a text block containing the regulatory text and a signature block for every author. The command is localised for both english and german. Using the optional argument, you can define the positioning. Pass `t` for alignment at the top of the page and `b` for bottom alignment (*default*).
- The three commands `\frontmatter`, `\mainmatter` and `\backmatter` are provided for report classes. They separete document section and automatically set up `pagenumer` styles.
- `\frontmatter` `\frontmatter` set the pagenumers to capital roman numerals. This is usually required for everything before the first chapter.
- `\mainmatter` `\mainmatter` sets the pagenumbering to "normal" arabic numbers. This is usually the style for the document content.
- `\backmatter` `\backmatter` can be used for apendixes and alike. It sets the pagenumbering to small roman numerals.

3.3 Default L^AT_EX Commands that are modified

Additionally, some of L^AT_EXs default commands are modified to better fit this class. This is documented here.

- `\title` The `\title[<short-title>]{<title>}` command now takes an additional, optional argument. You can use it for a shorter version of your title, that will be used in the header to save on space.
- `\maketitle` The `\maketitle` command if L^AT_EX default way to create a titlepage. We redefine it to produce a titlepage that matches the sytel typically used on the Hochschule Hannover. This includes the logo beeing printed, depending on the `f1` to `f5` documentoptions. The command now takes an options alignment-parameter: `\maketitle[<align>]`. You can pass `l` | `c` | `r` to get *left*, *center* or *right* alignment.

3.4 Micalanious commands

The following commands are mostly for internal use, but are parte of the interface and might be usefull in some cases.

- `\ifsingleauthor` This allows you to check if `\author` holds one or multiple entries. Using this allows the user to display different things for these cases, for example switch words to plural, like this:
`\ifsingleauthor{one}{multiple} person\ifsingleauthor{}{s}`

4 Package loading

The classes load some packages for internal use as well as loading and configuring common use packages. The details are documented in the following section.

4.1 Always loaded packages

fontenc for output encoding, set to the european character set
babel for language-specific typesetting
bookmark creates bookmarks in the PDF
hyperref for easy referencing and linking
caption to customize captions and make references point to the beginning of the floats
graphicx for importing and manipulating images
amsmath,amssymb,amsfonts more options when typesetting math
lmodern sets up the Latin-Modern font
setspace used for configuring linespacing

There are also some packages for internal functionality that shouldn't concern the user, but they are listed here for completeness.

scrbase
pgffor

4.2 Conditionally loaded Packages

A subset of packages is only loaded (or loaded with specific options) depending on options passed to the package

babel gets configured depending on language
csquotes for language-specific quotations marks
ziffer only loaded for german documents, sets comma as decimal separator
todonotes loaded depending on todos

4.3 Pre-configured Packages

These packages are configured by the class to work in a cooperative way. The user must load them in this preamble via `\usepackage{<pkg-name>}` however, as loading them always bears additional, unnecessary overhead.

5 Implementation

5.1 Internal commands

`\HsHClassName` The classname of specific class is stored in the `\HsHClassName` which gets used throughout the code.

```
1 \let\HsHClassName\@currname
```

There is also a second macro `\HsHClassName@ParentClass` which stores the parent class name.

```
2 \def\HsHClassName@ParentClass{%  
3 <article> scrartcl%  
4 <report> scrreprt%  
5 <standalone> standalone%  
6 }
```

5.2 Option handling

The options are handled using features provided by the KOMA-Script ecosystem. To get access to this the `srcbase` package is loaded.

```
7 \RequirePackage{srcbase}
```

We also require some packages for some of the option, these are loaded next.

```
8 \RequirePackage{setspace}
```

A new family of keys is created and shared by all elements of this project. For that the `\DefineFamily` macro is used. The family name is *HsH*, matching the usual prefixes.

```
9 \DefineFamily{HsH}
```

Additionally each class-file represents a member in the family. This is defined using the `\DefineFamilyMember` macro. It's optional argument is set to the current filename by default, so we do not need to specify it, just the family name to attach it to.

```
10 \DefineFamilyMember{HsH}
```

`\HsH@Options@PassToParent` To be able to pass options to the parent class where needed easily, a command is defined. It also issues a log-message.

```
11 \newcommand{\HsH@Options@PassToParent}[1]{%
12   \ClassInfoNoLine{\HsHClassName}{passing option to parent class: #1}%
13   \PassOptionsToClass{#1}{\HsHClassName@ParentClass}%
14 }
```

`\HsH@Options@DeclareAlias` It is also useful to have short-versions of options. The following commands make it easy to declare these.

```
15 \newcommand{\HsH@Options@DeclareAlias}[3][HsH]{%
16   \DeclareOption{#2}{\FamilyExecuteOptions{#1}{#3}}%
17 }
```

`fontfamily` The first option to be defined is `fontfamily`. It's defined as a *Numerical* key so that it can accept multiple options and map them to a switch case.

```
18 \DefineFamilyKey{HsH}{fontfamily}{
19   \begingroup
20   \FamilySetNumerical{HsH}{parskip}{@tmp}{%
21     {sans}{0}, {sans-serif}{0},%
22     {roman}{1}, {serif}{1},%
23   }{#1}
24   \ifx\FamilyKeyState\FamilyKeyStateProcessed
25     \aftergroup\FamilyKeyStateProcessed
26     \ifcase@tmp% 0
27       \endgroup
28       \renewcommand{\familydefault}{\sfdefault}
29       \if@atdocument\AfterKOMAoptions{\selectfont}\fi
30     \or% 1
31       \endgroup
32       \renewcommand{\familydefault}{\rmdefault}
33       \if@atdocument\AfterKOMAoptions{\selectfont}\fi
34     \else% should never be
35       \endgroup
36     \fi
37   \else
38     \endgroup
39     \FamilyKeyStateUnknownValue
40   \fi
41 }
```

`sans` For convenience, there are also two short-versions defined.

```
roman 42 \HsH@Options@DeclareAlias{sans}{fontfamily=sans}
43 \HsH@Options@DeclareAlias{roman}{fontfamily=roman}
44 \!*standalone
```

`linespacing` The `linespacing` options is also a *Numerical* option, mapping to three cases. They execute the appropriate commands of the `setspace` package.

```

45 \DefineFamilyKey{HsH}{linespacing}{%
46   \begingroup
47   \FamilySetNumerical{HsH}{linespacing}{@tempa}{%
48     {single}{0},%
49     {onehalf}{1},%
50     {double}{2},%
51   }{#1}
52   \ifx\FamilyKeyState\FamilyKeyStateProcessed
53     \aftergroup\FamilyKeyStateProcessed
54     \ifcase \@tempa% 0
55       \endgroup
56       \if@atdocument\singlespacing\else\AtEndOfClass{\singlespacing}\fi
57     \or% 1
58       \endgroup
59       \if@atdocument\onehalfspacing\else\AtEndOfClass{\onehalfspacing\AfterTOCHead{\singlespacing}}\fi
60     \or% 2
61       \endgroup
62       \if@atdocument\doublespacing\else\AtEndOfClass{\doublespacing\AfterTOCHead{\singlespacing}}\fi
63
64     \else% should never be
65       \endgroup
66     \fi
67   \else
68     \endgroup
69     \FamilyKeyStateUnknownValue
70   \fi
71 }
```

`singlespacing` For convenience, there are also these short-versions defined.

```

onehalfspacing 72 \HsH@Options@DeclareAlias{singlespacing}{linespacing=single}
doublespacing 73 \HsH@Options@DeclareAlias{onehalfspacing}{linespacing=onehalf}
               74 \HsH@Options@DeclareAlias{doublespacing}{linespacing=double}
```

`parskip` The `parskip` option is special in that it originally a KOMA-Script option that get expanded by this class. Only two new cases are defined here and everything unknown gets passed to the parent class.

It should also be noted that this option can't execute its code immediately, as the commands needed are only defined later when the parent class loads in. So the `\setparsizes` command is pushed into a hook.

```

75 \DefineFamilyKey{HsH}{parskip}{%
76   \begingroup
77   \FamilySetNumerical{HsH}{parskip}{@tempa}{%
78     {never+}{0},%
79     {never*}{1},%
80   }{#1}
81   \if@atdocument
82     \ClassError{\HsHClassName}{
83       option 'parskip' can only be configured in preamble!
84     }
85   \fi
86   \ifx\FamilyKeyState\FamilyKeyStateProcessed
87     \aftergroup\FamilyKeyStateProcessed
88     \ifcase \@tempa% 0
89       \endgroup
90       \AtEndOfClass{\setparsizes{\z@}{\z@}{.3333\linewidth \@plus 1fil}}
91     \or% 1
92       \endgroup
93       \AtEndOfClass{\setparsizes{\z@}{\z@}{.25\linewidth \@plus 1fil}}
94     \else% should never be
95       \endgroup
96     \fi
```

```

97  \else
98  \endgroup
99  \HsH@Options@PassToParrent{parskip=#1}
100 \FamilyKeyStateProcessed
101 \fi
102 }

```

headheight The headheight option just set the \headheight to the given value.

```

103 \def\HsH@opt@headheight{}
104 \FamilyStringKey{HsH}{headheight}{\HsH@opt@headheight}
105 \AtEndOfClass{%
106   \headheight=\HsH@opt@headheight%
107 }

```

abstract The abstract option sets multiple different switches and configurations. We first define the needed macros:

```

108 \newif\if@HsH@option@abstract@show@keywords
109 \def\HsH@abstract@heading{%
110   <article> \addsec%
111   <report> \addchap%
112 }

```

Now the actual option can be defined to handle all the cases.

```

113 \DefineFamilyKey{HsH}{abstract}{%
114   \begingroup
115   \FamilySetNumerical{HsH}{abstract}{@tempa}{%
116     {keywords}{0},%
117     {nokeywords}{1},%
118     {totoc}{2},{toc}{2},%
119     {notoc}{3},{nottotoc}{3},%
120   }{#1}
121   \ifx\FamilyKeyState\FamilyKeyStateProcessed
122     \aftergroup\FamilyKeyStateProcessed
123     \ifcase\@tempa% 0
124       \endgroup
125       \@HsH@option@abstract@show@keywordstrue
126     \or% 1
127       \endgroup
128       \@HsH@option@abstract@show@keywordsfalse
129     \or% 2
130       \endgroup
131       \def\HsH@abstract@heading{%
132 <article> \addsec%
133 <report> \addchap%
134       }
135     \or% 3
136       \endgroup
137       \def\HsH@abstract@heading{%
138 <article> \addsec%
139 <report> \addchap%
140       }
141     \else% should never be
142       \endgroup
143     \fi
144   \else
145     \endgroup
146   \fi
147 }

```

toc The toc option is a KOMA-Script option we just extend. Options for the abstract and toc are added.

```

148 \DefineFamilyKey{HsH}{toc}{%
149   \begingroup
150   \FamilySetNumerical{HsH}{toc}{@tempa}{%

```

```

151 {totoc}{0},{toc}{0},%
152 {notoc}{1},{nottotoc}{1},%
153 {abstract}{2},%
154 {noabstract}{3},%
155 }{#1}
156 \ifx\FamilyKeyState\FamilyKeyStateProcessed
157 \aftergroup\FamilyKeyStateProcessed
158 \ifcase\@tempa% 0
159 \endgroup
160 \AtEndOfClass{\setuptoc{toc}{totoc}}
161 \or% 1
162 \endgroup
163 \AtEndOfClass{\unsettoc{toc}{totoc}}
164 \or% 2
165 \endgroup
166 \FamilyExecuteOptions{HsH}{abstract=totoc}
167 \or% 3
168 \endgroup
169 \FamilyExecuteOptions{HsH}{abstract=nottotoc}
170 \else% should never be
171 \endgroup
172 \fi
173 \else
174 \endgroup
175 \HsH@Options@PassToParrent{toc=#1}
176 \FamilyKeyStateProcessed
177 \fi
178 }

```

For the twoside option we only redefine the default, everything else is handled by the parent class.

```

179 \DefineFamilyKey{HsH}{twoside}[semi]{%
180 \HsH@Options@PassToParrent{twoside=#1,BCOR=1cm}
181 \FamilyKeyStateProcessed
182 }
183 </!standalone>

```

`\HsH@opt@language` First, the macro to store the languages name in is created. The default is `nil`, as babel will see this as no-language.

```
184 \def\HsH@opt@language{nil}
```

`language` The key is then defined to store its value inside the command. This allows the option to be called multiple time, but only the last set value will be passed on to babel.

```
185 \FamilyStringKey{HsH}{language}{\HsH@opt@language}
```

`english` For convenience, there are also these short-versions defined.

```

german 186 \HsH@Options@DeclareAlias{english}{language=english}
ngerman 187 \HsH@Options@DeclareAlias{german}{language=ngerman}
188 \HsH@Options@DeclareAlias{ngerman}{language=ngerman}

```

`\HsH@opt@faculty` As we need a default value that is not zero, the macro needs to be defined an initialised manually.

```
189 \def\HsH@opt@faculty{1}
```

`faculty` The faculty options is once again a *Numerical* option, mapping the five faculties and storing the selected one in `\HsH@opt@faculty`.

```

190 \FamilyNumericalKey{HsH}{faculty}{HsH@opt@faculty}{%
191 {0}{1}, {none}{1}, {false}{1}, {off}{1},%
192 {1}{2}, {f1}{2},%
193 {2}{3}, {f2}{3},%
194 {3}{4}, {f3}{4},%
195 {4}{5}, {f4}{5},%

```

```

196 {5}{6}, {f5}{6},%
197 }

```

f1 For convinience, there are also these short-versions defined.

```

f2 198 \HsH@Options@DeclareAlias{f1}{faculty=f1}
f3 199 \HsH@Options@DeclareAlias{f2}{faculty=f2}
f4 200 \HsH@Options@DeclareAlias{f3}{faculty=f3}
f5 201 \HsH@Options@DeclareAlias{f4}{faculty=f4}
    202 \HsH@Options@DeclareAlias{f5}{faculty=f5}

```

todos The boolean option todos is simply created using the commands from scrbase. Boolean options allready default to `<true>` if called without and argument, so no need to define an explicit alias.

```

203 \FamilyBoolKey{HsH}{todos}{@todos}

```

For the standalone class the fontsize option is mocked to present a standardised interface. A user might expect this option to be passable to this class and we shouldnd crete an anoying error just for this.

```

204 <standalone>
205 \DefineFamilyKey{HsH}{fontsize}{%
206   \ClassInfoNoLine{\HsHClassName}{The 'fontsize' option is only a mock, its has not effect}
207   \FamilyKeyStateProcessed
208 }
209 </standalone>

```

5.2.1 Unknown options

Unknown options will be passed to the parent class. For that a `@else@` key is defined on the HsH family, which will be execute for every unknown key-value option. Unknown bare options are handled by the `\DeclareOption*` command and will be passed there.

```

210 \DefineFamilyKey{HsH}{@else@}{
211   \HsH@Options@PassToParrent{#1}
212   \FamilyKeyStateProcessed
213 }
214 \DeclareOption*{
215   \HsH@Options@PassToParrent{\CurrentOption}
216 }

```

5.2.2 Default options

The different classes all execute a set of default options, which is handled by the following code.

```

217 \FamilyExecuteOptions{HsH}{%
218   fontfamily=sans-serif,
219 <!*standalone>
220   fontsize=11pt,
221   language=ngerman,
222 <article> parskip=never+,
223 <report> parskip=half+,
224   linespacing=single,
225   headheight=2.15\baselineskip,
226 </!standalone>
227 <*article | report>
228   toc=listof,
229   toc=bibliography,
230   abstract=keywords,
231 </article | report>
232   faculty=none,
233 <*standalone>
234   margin=0.25cm,
235   multi=tikzpicture,
236   multi=circuitikz,

```

```

237 </standalone>
238 }

```

Now we can process the options for the HsH family.

```

239 \FamilyProcessOptions{HsH}\relax

```

5.2.3 Loading the parent class

```

240 \LoadClass{\HsHClassName@ParrentClass}

```

5.3 Package loading

The classes load and configure some common packages to reduce the needed amount of boilerplate code in your preamble.

Additionally there are settings provided for packages that are used more rarely, but will be set up correctly if you decide to load them via `\usepackage{}`.

5.3.1 Ensuring German works

With modern LaTeX systems the encoding of inputfiles is UTF-8 by default, so the inputenc package is no longer required. Should the user still use a old setup or use a different encoding, he is responsible for loading inputenc himself.

The font-encoding for the pdf file is also set up to allow for the full european character set.

```

241 \RequirePackage[T1]{fontenc}
242 \RequirePackage{type1ec}

```

To ensure localised translations of all displayed text automatically depend on the user-selected language, the babel package is loaded. This also allows for the use of the `\iflanguage` command, which is relevant later.

```

243 \RequirePackage[main=\HsH@opt@language]{babel}

```

Quotation marks are also very different between languages, so the following ensures the correct style for the correct language.

```

244 \RequirePackage[autostyle=true]{csquotes}
245 \MakeOuterQuote{"}

```

German uses a comma as the decimal separator, which collides with LaTeX's default english setting of using the comma as a thousands separator and therefore replacing it with some whitespace on printed version. Luckily loading the ziffer package sets this up to match the german standard.

```

246 \iflanguage{ngerman}{\RequirePackage{ziffer}}{}

```

5.3.2 Generally useful packages

We load hyperref for clickable links and configure it to write meta-data to the PDF.

```

247 \RequirePackage[hidelinks]{hyperref} % must load before 'bookmarks'
248 \RequirePackage{bookmark}
249 <!*standalone>
250 \AtBeginDocument{
251   \hypersetup{
252     pdfinfo={
253       Title={\@title},
254       Author={\@author},
255       Subject={\@subject},
256       Keywords={\@keywords}
257     }
258   }
259 }
260 </!standalone>

```

The todonotes package is great for annotation, but extremely expensive on compiletime. So we load it only if the user requests it. Also its commands are stubbed, so that they can be left in the sourcecode and just not output anything.

```

261 \if@todos

```

```

262 \PassOptionsToPackage{
263     textsize=small,
264     figwidth=.6\textwidth
265 }{todonotes}
266 \RequirePackage{todonotes}
267 \else
268 \newcommand{\listoftodos}[1]{%
269 \newcommand{\todo}[2] [] {}
270 \newcommand{\missingfigure}[2] [] {}
271 \fi

272 \RequirePackage[hypcap=true]{caption}
273 \RequirePackage{graphicx}
274 \RequirePackage{amsmath,amssymb,amsfonts}
275 \RequirePackage{svgnames}{xcolor}

```

5.3.3 Options for packages that could be loaded by the user

Some package are not always needed and potentially heavy to load in by default. But its still usefull to set default options for these packages.

These differ from the settings provided in `HsH-classes.cfg` in that they are defaults that apply allway and not user-configurable preferences which are user or even project specific. For the bibtex we ensure the *biber* backend is selcted, which matches the settings in `.latexmkrc`.

```

276 (*article | report)
277 \PassOptionsToPackage{backend=biber}{biblatex}
278 \AtBeginDocument{
279 \makeatletter
280 \@ifpackageloaded{biblatex}{
281 \renewcommand*{\mkbibacro}[1]{\MakeUppercase{#1}}
282 }{}%
283 \makeatother
284 }
285 \end{article}

```

For bibtex we load the free-standing units, mostly for backwards compatibility. We also ensure german language specific settings are applied.

```

286 \PassOptionsToPackage{free-standing-units}{siunitx}
287 \AtBeginDocument{
288 \makeatletter
289 \@ifpackageloaded{siunitx}{
290 \iflanguage{ngerman}{
291 \sisetup{output-decimal-marker={,}}
292 }{}
293 }{}
294 \makeatother
295 }

```

For better compatibility with the listings package we load the scrhack package. We also pass some configurations to if it gets loaded.

```

296 \RequirePackage{scrhack}
297 \AtBeginDocument{
298 \makeatletter
299 \@ifpackageloaded{listings}{
300 \RequirePackage{lstautogobble}\lstset{autogobble=true}
301 \iflanguage{ngerman}{
302 \lstset{iterate={0}{\ "0}}1{Ä}{\ "A}}1{Ü}{\ "U}}1{ß}{\ "ss}}1{ü}{\ "u}}1{ä}{\ "a}}1{ö}{\ "o}}
303 }{}
304 }{}
305 \makeatother
306 }

```

The circuitikz needs bo be configure so it matches typical european styles.

```

307 \PassOptionsToPackage{european,EFvoltages,straightvoltages,betterproportions}{circuitikz}

```

For other packages we provide the settings more as a recommendation of what is usefull. As the user might want to change these, we outsource this to a seperat file and input it `\AtBeginDocument`. That way the user can just replace the file with his custom version.

```

308 \AtBeginDocument{
309   \makeatletter
310   \InputIfFileExists{HsH-classes.cfg}{
311     \ClassInfo{\HsHClassName}{Local config file HsH-classes.cfg used.}
312   }{
313     \ClassInfo{\HsHClassName}{No HsH-classes.cfg!! I hope you configered it yourself.}
314   }
315   \makeatother
316 }

```

5.4 Custom commands

5.4.1 Document seperation commands

The following commands are only defnied for book type classes by default. But they are also usefull for the report class, so we define them in that case.

```
317 <{*report}
```

`\if@mainmatter` We define a switch which stores if the document is currently at a mainmatter section. It defaults to true as the user needs to explicitly set the state to something differnt.

```
318 \newif\if@mainmatter\@mainmattertrue
```

As a pagenumber change requires a fresh page, this is ensured first. We also need to make sure that on twosided document, the first page is always on the left.

`\frontmatter` The pagenumbering is set to capital roman numerals.

```

319 \newcommand{\frontmatter}{
320   \if@twoside\cleardoubleoddpage\else\clearpage\fi
321   \@mainmatterfalse\pagenumbering{Roman}
322 }

```

`\mainmatter` The pagenumbering is set to arabic numerals.

```

323 \newcommand{\mainmatter}{
324   \if@twoside\cleardoubleoddpage\else\clearpage\fi
325   \@mainmattertrue\pagenumbering{arabic}
326 }

```

`\backmatter` The pagenumbering is set to arabic numerals.

```

327 \newcommand{\backmatter}{
328   \if@openright\cleardoubleoddpage\else\clearpage\fi
329   \@mainmatterfalse\pagenumbering{roman}
330 }

```

```
331 </report>
```

5.4.2 The Logo for Hochschule Hannover

The following macros are responsible for creating the logo. They load a specific page of a PDF file and siplay it.

`\HsHlogoPath` This macro contains the path to load the PDF from. It defaults to HSH-Logo.pdf, which is provided by this project inside the `scr/` folder.

```
332 \newcommand{\HsHlogoPath}{HSH-Logo.pdf}
```

`\HsHlogoPage` This macro stores the page to use from the PDF. It will be set via the `documentoption` faculty.

```
333 \newcommand{\HsHlogoPage}{\HsH@opt@faculty}
```

`\includeHsHlogohere` Calling this macro produces the logo in-place. You can specify the width as an optional argument. The default is 5cm.

If the file provided via `\HsHlogoPath` doesn't exist, the command will produce an error.

```

334 \newcommand{\includeHsHlogohere}[1][5cm]{
335   \IfFileExists{\HsHlogoPath}{
336     \includegraphics[width=#1,page=\HsHlogoPage]{\HsHlogoPath}
337   }{
338     \ClassError{\HsHClassName}{\HsHlogoPath\space not found!}{
339       The HsH Logo is necessary for the titlepage! Try putting it next to your source file or u
340     }
341   }
342 }
```

5.4.3 Title matters

343 `<{*article | report}`

The following commands relate to the creation of the titlepage. They implement how the user can define the different datafields.

`\title` We redefine the `\title` command to take an optional argument. This is stored in the additional `\@shorttitle` macro.

```

344 \renewcommand{\title}[2][]{
345   \gdef\@title{#2}
346   \gdef\@shorttitle{#1}
347 }
```

`\@shorttitle` This new macro stores a short version of the title. This will be used in places where the full title might overflow the available space.

```
348 \def\@shorttitle{\@empty}
```

`\gdef@with@and@as@comma` First we define a helpermacro that will fill a given macro while treating the `\and` command as a comma. This is needed for `\author` and `\matrikelnr` where we later treat them as comma-separated lists when the titlepage gets constructed.

```
349 \def\gdef@with@and@as@comma#1#2{{\def\and{,}\xdef#1{#2}}}
```

`\author` We redefine the `\author` command so that it sets `\@author`, but allowing for both comma and `\and` to be used as a separator.

```
350 \renewcommand{\author}[1]{\gdef@with@and@as@comma\@author{#1}}
```

The `\@author` macro is also set to `\@empty`, this makes it easier to handle it later.

```
351 \let\@author\@empty
```

`\matrikelnr` These macros set and store the matrikel-number (or set of numbers), which will be printed on the titlepage.

```

352 \newcommand{\matrikelnr}[1]{\gdef@with@and@as@comma\@matrikelnr{#1}}
353 \def\@matrikelnr{\@empty}
```

`\professor` These three macros-groups give options to the user to print people's names on the titlepage, who are relevant to the document, but not the author.

```

\secondexaminer 354 \newcommand{\professor}[1]{\gdef\@professor{#1}}
\@professor      355 \def\@professor{\@empty}
\@firstexaminer  356 \newcommand{\firstexaminer}[1]{\gdef\@firstexaminer{#1}}
\@secondexaminer 357 \def\@firstexaminer{\@empty}
                 358 \newcommand{\secondexaminer}[1]{\gdef\@secondexaminer{#1}}
                 359 \def\@secondexaminer{\@empty}
```

`\keywords` The macro-group defines and holds keywords which describe the document. They are used when printing the abstract as well as in the PDF's meta-data.

```

360 \newcommand{\keywords}[1]{\gdef\@keywords{#1}}
361 \def\@keywords{\@empty}
```

```
362 </article | report>
```

5.4.4 Commands for document writing

`\declarationofauthorship` The declaration of authorship is not relevant for the standalone variant.

```
363 \*!stadnalone
```

The `pgffor` package is required to handle the loop over the list of authors.

```
364 \RequirePackage{pgffor}
```

Now the command is defined. It takes an optional argument which defaults to `b`.

```
365 \newcommand{\declarationofauthorship}[1][b]{
```

First the argument is passed and an error raised for invalid arguments. Passing in `b` will push the declaration to the bottom of the page and add a horizontal line. Passing `t` simply adds some space.

```
366 \if#1b
```

```
367 \vspace*{\fill}
```

```
368 \hrule
```

```
369 \else\if#1t
```

```
370 \vspace*{2em}
```

```
371 \else
```

```
372 \ClassError{\HsHClassName}{Wrong Parameter for '\declarationofauthorship'}{
```

```
373 '\string\declarationofauthorship' only accepts 't' and 'b' as parameters.
```

```
374 }
```

```
375 \fi\fi
```

Now the actual declaration can be constructed. It uses the text from `\decofauthname` and `\decofauthtext`.

```
376 \vskip 3em
```

```
377 {\centering\bfseries\usekomafont{section}{\decofauthname}\par}
```

```
378 \vskip 3em
```

```
379 \decofauthtext\par
```

The last step is to loop over all authors by reading `\@author` and creating a signature box for each one. `\thanks` also needs to be cleared, as a footnote wouldn't make sense here.

```
380 \begingroup
```

```
381 \renewcommand{\thanks}{\sbox0{
```

```
382 \raggedleft
```

```
383 \foreach \tmp@ in \@author {
```

```
384 \if\tmp@empty\else
```

```
385 \hskip 1em \parbox{4cm}{
```

```
386 \vskip 4em
```

```
387 \hrule\vskip 4pt
```

```
388 \raggedleft\footnotesize\tmp@
```

```
389 }%
```

```
390 \fi
```

```
391 }\par
```

```
392 \endgroup
```

```
393 }
```

`\if@singleauthor` To ensure `\decofauthtext` is properly spelled for one or multiple authors, we define a conditional that holds this information. Additionally we check the number of authors `\AtBeginDocument` and store it.

```
394 \newif\if@singleauthor
```

```
395 \AtBeginDocument{
```

```
396 \begingroup
```

```
397 \newcount\count@
```

```
398 \count@=\z@
```

```
399 \@for\tmp@:=\@author\do{\advance\count@\@ne}
```

```
400 \ifnum\count@>\@ne\global\@singleauthorfalse\else\global\@singleauthortrue\fi
```

```
401 \endgroup
```

```
402 }
```

`\ifsingleauthor` A macro to simply act on this condition is also provided. It will be used in text-definitions later on.

```
403 \newcommand{\ifsingleauthor}[2]{\if@singleauthor#1\else#2\fi}
```

`\declarationAuthorship`

```
404 \def\declarationAuthorship{%
405   \ClassWarning{\HsHClassName}{%
406     Command \string\declarationAuthorship\space is deprecate.\MessageBreak
407     Replace it with \string\declarationofauthorship.
408   }%
409   \declarationofauthorship%
410 }
411 \<!/standalone>
```

5.4.5 Micalanious commands

Utility commands

For writing absolout values, we provide the `\abs{ $\langle equ \rangle$ }` command, which puts groable, vertical bars on both sides of the equation inside.

```
412 \newcommand{\abs}[1]{\ensuremath{\left\vert\right\vert#1\right\vert}}
```

Configuring mathmode-indices

The only hard requirements for documents writing on Hochschule Hannover is, that the indices in mathematic formulas must be typset in an upright ("steil") font, not the default kursive font. We configure this by first defining a macro to replace the default `\sb` macro. We can than assign this to `_`. For that to work we need to change its catcode to make it modifyable.

Note: You can allways use `\sb` to use the original behaviour for special cases.

```
413 \def\@subinrm#1{\sb{\mathrm{#1}}}%
414 {\catcode'\_ =13 \global\let\_ =\@subinrm}
```

`\upsubscripts` Now we can define a command to activate this new behavior by changing the catcode of `_` to 13, which makes it a normal macro.

```
415 \newcommand\upsubscripts{\catcode'\_ =12}
```

`\normalsubscripts` To switch back we simply need to reset the catcode of `_` back to the original, which makes it a builidin operator with the default behavior.

```
416 \newcommand\normalsubscripts{\catcode'\_ =8}
```

5.5 Document setup

The following sets up the look and feel of the documents using this classe. All configuration and stylin is done here.

5.5.1 Fonts and text styling

```
417 \RequirePackage{lmodern}
```

5.5.2 Page layout

```
418 \<!*standalone>
419 \areaset[current]{0.75\paperwidth}{0.8\paperheight}
420 \if@todos
421   \addtolength\paperwidth{5cm}
422   \addtolength\marginparwidth{5cm}
423 \fi
424 \<!/standalone>
```

5.5.3 Styling L^AT_EX default constucts

Floats

```
425 \<!*standalone>
```

Floats should alwas prefer the *here* placement, than the *top* of the following page.

```
426 \renewcommand{\fps@figure}{h!t}
427 \renewcommand{\fps@table}{h!t}
```

Floats should be centered by default and the width of the caption box is limited.

```
428 \g@addto@macro\@floatboxreset\centering
429 \setcapwidth{0.8\textwidth}
```

The names of floating environments are redefined to show abbreviations only.

```
430 \defcaptionname{english}\figurename{Fig.}
431 \defcaptionname{german,ngerman}\figurename{Abb.}
432 \defcaptionname{english}\tablename{Tab.}
433 \defcaptionname{german,ngerman}\tablename{Tab.}
```

For subfigures we need to define a name used in autoreferences.

```
434 \AtBeginDocument{
435   \makeatletter
436   \@ifpackageloaded{subfigure}{
437     \let\subfigureautorefname\figureautorefname
438   }{}%
439   \makeatother
440 }
441 \</!standalone>
```

Lists

For unordert liste the markers are redefined to look a littel nicer.

```
442 \renewcommand{\labelitemi}{\raisebox{.3ex}{\scalebox{0.7}{\bullet$}}}
443 \renewcommand{\labelitemii}{\raisebox{.3ex}{\scalebox{0.7}{\circ$}}}
444 \renewcommand{\labelitemiii}{\raisebox{.1ex}{-}}
445 \renewcommand{\labelitemiv}{\raisebox{-.1ex}{\scalebox{1.3}{\cdot$}}}
```

Abstract

We define some custom behavior for the abstract.

```
446 \<!*standalone>
447 \renewenvironment{abstract}{
448   \quotation
449   \setparsizes{\z@}{\z@}{.25\linewidth \@plus 1fil}\selectfont
450   \HsH@abstract@heading{\abstractname}
451 }{}%
452   \ifx\@keywords\@empty\else\if@HsH@option@abstract@show@keywords
453     \par\bigskip
454     \noindent\textbf{\keywordsname}\hskip 2em\@keywords
455   \fi\fi\par
456   \endquotation
457 }
458 \</!standalone>
```

Misc

We activate one of the commands defined above to make math-indices upright by default.

```
459 \upsubscripts
```

We want a ragged bottom instead of spreading the paragraphs over the page.

```
460 \raggedbottom
```

The ruler shown in the top and left margin with the draft option is removed.

```
461 \let\layercontentsmeasure\relax
```

5.5.4 Header and footer

```
462 \<article | report>
```

The header and footer are styled using the low-level commands provided by the KOMA-Script package `scrlayer-scrpage`.

```
463 \RequirePackage{scrlayer-scrpage}
464 \FamilyOptions{KOMA}{headsepline,singlespacing=true}
```

First we define the new pagestyle `HsHheadings`.

```

465 \newpagestyle{HsHheadings}{
466   {
467     \parbox[b]{\sls@headwidth}{
468       \LaTeXraggedright
469       \ifx\@shorttitle\@empty\@title\else\@shorttitle\fi
470     }%
471   }
472   {
473     \parbox[b]{\sls@headwidth}{
474       \LaTeXraggedleft
475       \leftmark
476     }%
477   }
478   {
479     \parbox[b]{.45\sls@headwidth}{
480       \LaTeXraggedright
481       \ifx\@shorttitle\@empty\@title\else\@shorttitle\fi
482     }%
483     \hfill
484     \parbox[b]{.45\sls@headwidth}{
485       \LaTeXraggedleft
486       \headmark
487     }%
488   }
489   (\textwidth,.1pt)
490 }{
491   {\pagemark}
492   {\hfill\pagemark}
493   {\hfill\pagemark}
494 }

```

Then all generic settings are applied:

```

495 \clearpairofpagestyles
496 \ofoot*{\pagemark}
497 \pagestyle{HsHheadings}
498 \article\automark{section}
499 \report\automark{chapter}
500 \report\renewcommand*{\chapterpagestyle}{HsHheadings}
501 \</article | report>

```

5.5.5 Titlepage

`\maketitle` The definition of `\maketitle` is mostly taken from the source-code of the KOMA-Script parentclass, but was modified to create the desired style.

```

502 \<article | report>
503 \newtoks\@tabtoks
504 \newcommand\addtabtoks[1]{\global\@tabtoks\expandafter{\the\@tabtoks#1}}
505 \newcommand\eadddtabtoks[1]{\edef\mytmp{#1}\expandafter\addtabtoks\expandafter{\mytmp}}
506 %%%\newcommand*\resettabtoks{\global\@tabtoks{}}
507 \newcommand*\printtabtoks{\the\@tabtoks}
508 \addtokomafont{publishers}{\normalsize}
509 \g@addto@macro\titlepage{\singlespacing}
510 %
511 \article\renewcommand\maketitle[1][c]{
512 \report\renewcommand\maketitle[1][l]{
513   \expandafter\ifnum \csname scr@v3.12\endcsname>\scr@compatibility\relax
514   \else
515     \def\and{%
516       \end{tabular}
517       \hskip 1em \@plus.17fil
518       \begin{tabular}[t]{c}%
519     }
520   \fi
521 \<article>
522   \par

```

```

523 \ifx\@uppertitleback\@empty\else
524 \ClassWarning{\KOMAClassName}{%
525 non empty \string\uppertitleback\space ignored
526 by \string\maketitle\MessageBreak
527 in 'titlepage=false' mode%
528 }
529 \fi
530 \ifx\@lowertitleback\@empty\else
531 \ClassWarning{\KOMAClassName}{%
532 non empty \string\lowertitleback\space ignored
533 by \string\maketitle\MessageBreak
534 in 'titlepage=false' mode%
535 }
536 \fi
537 \</article>
538 \<report> \begin{titlepage}
539 \<article> \begin{group
540 \let\@param#1
541 \ifx\@param\@empty
542 \ClassError{myClass}{\maketitle\space with empty option}{
543 \maketitle[] has been called (with an empty parameter), this doesn't work.
544 Use \maketitle instead.
545 }
546 \fi
547 \<report>
548 \if@titlepageiscoverpage
549 \edef\titlename@restore{
550 \noexpand\endgroup
551 \noexpand\global\noexpand\@colht\the\@colht
552 \noexpand\global\noexpand\@colroom\the\@colroom
553 \noexpand\global\vsizethe\vsizethe
554 \noexpand\global\noexpand\@titlepageiscoverpagefalse
555 \noexpand\let\noexpand\titlename@restore\noexpand\relax
556 }
557 \begin{group
558 \topmargin=\dimexpr \coverpagetopmargin-1in\relax
559 \oddsidemargin=\dimexpr \coverpageleftmargin-1in\relax
560 \evensidemargin=\dimexpr \coverpageleftmargin-1in\relax
561 \textwidth=\dimexpr
562 \paperwidth-\coverpageleftmargin-\coverpagerightmargin\relax
563 \textheight=\dimexpr
564 \paperheight-\coverpagetopmargin-\coverpagebottommargin\relax
565 \headheight=0pt
566 \headsep=0pt
567 \footskip=\baselineskip
568 \colht=\textheight
569 \colroom=\textheight
570 \vsizethe\textheight
571 \columnwidth=\textwidth
572 \hsizethe\columnwidth
573 \linewidth=\hsizethe
574 \else
575 \let\titlename@restore\relax
576 \fi
577 \let\footnotesize\small
578 \let\footnoterule\relax
579 \let\footnote\thanks
580 \</report>
581 \<article> \let\titlename@restore\relax
582 \renewcommand*\thefootnote{\@fnsymbol\c@footnote}%
583 \let\@oldmakefnmark\@makefnmark
584 \renewcommand*\@makefnmark{\rlap\@oldmakefnmark}%
585 \<article> \next@tdpage

```

```

586 \ifx\@extratitle\@empty
587 <article> \ifx\@frontispiece\@empty\else \mbox{}\fi
588 <*report>
589 \ifx\@frontispiece\@empty\else
590 \if@twoside\mbox{}\next@tpage\fi
591 \noindent\@frontispiece\next@tdpage
592 \fi
593 </report>
594 \else
595 <article> \@makeextratitle
596 <*report>
597 \noindent\@extratitle
598 \ifx\@frontispiece\@empty
599 \else
600 \next@tpage
601 \noindent\@frontispiece
602 \fi
603 \next@tdpage
604 </report>
605 \fi
606 <*article>
607 \ifx\@frontispiece\@empty
608 \ifx\@extratitle\@empty\else\next@tdpage\fi
609 \else
610 \next@tpage
611 \@makefrontispiece
612 \next@tdpage
613 \fi
614 \global\@topnum=\z@
615 </article>
616 \setparsizes{\z@}{\z@}{\z@\@plus 1fil}\par@updaterelative
617 \vspace*{1cm}
618 \begin{minipage}[t]{\textwidth}%
619 \ifx\@titlehead\@empty \else
620 \usekomafont{titlehead}{\@titlehead}%
621 \fi
622 \hfill
623 % image with referencepoint in lower left corner:
624 \raisebox{0pt}[\ht\strutbox][\dp\strutbox]{\includeHsHlogohere}
625 \end{minipage}
626 \raisebox{10pt}{\rule{\textwidth}{0.5pt}}
627 \null
628 <article> \vskip 2em
629 <report> \vfill
630 \begingroup
631 \if\@param c\centering\fi
632 \if\@param r\raggedleft\fi
633 \ifx\@subject\@empty\else
634 {\usekomafont{subject}{\@subject\par}}
635 <article> \vskip 1.5em
636 <report> \vskip 3em
637 \fi
638 {\usekomafont{title}{\huge\@title\par}}
639 <article> \vskip .5em
640 <report> \vskip 1em
641 {\ifx\@subtitle\@empty\else\usekomafont{subtitle}\@subtitle\par\fi}
642 <article|report> \vskip 4em
643 {\ifx\@matrikelnr\@empty
644 \if\@author\@empty\else\usekomafont{author}{
645 \parbox{\dimexpr\linewidth}{
646 \if\@param c\centering\fi
647 \if\@param r\raggedleft\fi
648 \@author

```

```

649     }
650   }\fi
651   \else
652     \if\@author\@empty\else
653       % sneaky comma needed after \@matrikelnr to deal with single item lists
654       \foreach \x [count=\i,evaluate=\i as \y using {\@matrikelnr,\i-1}] in \@author {
655         \usekomafont{author}{
656           \def\arraystretch{1.2}
657           \if\@param l\begin{tabular}{@{}l l}\fi
658           \if\@param c\begin{tabular}{l l}\fi
659           \if\@param r\begin{tabular}{r r@{}}\fi
660           \printtabtoks
661           \end{tabular}
662         }%
663       \fi
664     \fi}
665   <article> \vskip 1em
666   <report> \vskip 1.5em
667   {\usekomafont{date}{\@date\par}}
668   <article> \vskip 1em
669   <report> \vskip \z@ \@plus3fill
670   \usekomafont{publishers}{
671     \def\arraystretch{1.2}
672     \if\@param l\begin{tabular}{@{}l l}\fi
673     \if\@param c\begin{tabular}{l l}\fi
674     \if\@param r\begin{tabular}{r r@{}}\fi
675     \if\@professor\@empty\else\textbf{\professorname:}&\@professor\\\fi
676     \if\@firstexaminer\@empty\else\textbf{\firstexaminername:}&\@firstexaminer\\\fi
677     \if\@secondexaminer\@empty\else\textbf{\secondexaminername:}&\@secondexaminer\\\fi
678     \end{tabular}
679   }
680   <*article>
681     \ifx\@dedication\@empty\else
682       \vskip 2em
683       {\usekomafont{dedication}{\@dedication \par}}%
684     \fi
685   </article>
686   \par
687   \endgroup
688   <article> \vskip 2em
689   <report> \vskip 3em
690   <article> \ifx\titlepagestyle\@empty\else\thispagestyle{\titlepagestyle}\fi
691   \@thanks\global\let\@thanks\@empty
692   <*report>
693     \vfill\null
694     \if@twoside
695       \@tempwatrue
696       \expandafter\ifnum \@nameuse{scr@v03.12}>\scr@compatibility\relax
697       \else
698         \ifx\@uppertitleback\@empty\ifx\@lowertitleback\@empty
699           \@tempswafalse
700         \fi\fi
701       \fi
702       \if@tempswa
703         \next@tpage
704         \begin{minipage}[t]{\textwidth}
705           \@uppertitleback
706         \end{minipage}\par
707       \vfill
708       \begin{minipage}[b]{\textwidth}
709         \@lowertitleback
710       \end{minipage}\par
711       \@thanks\global\let\@thanks\@empty

```

```

712     \fi
713   \else
714     \ifx\@uppertitleback\@empty\else
715       \ClassWarning{\KOMAClassName}{%
716         non empty \string\uppertitleback\space ignored
717         by \string\maketitle\MessageBreak
718         in 'twoside=false' mode%
719       }
720     \fi
721     \ifx\@lowertitleback\@empty\else
722       \ClassWarning{\KOMAClassName}{%
723         non empty \string\lowertitleback\space ignored
724         by \string\maketitle\MessageBreak
725         in 'twoside=false' mode%
726       }
727     \fi
728   \fi
729   \ifx\@dedication\@empty
730   \else
731     \next@tdpage\null\vfill
732     {\centering\usekomafont{dedication}{\@dedication \par}}%
733     \vskip \z@ \@plus3fill
734     \@thanks\global\let\@thanks\@empty
735     \cleardoubleemptypage
736   \fi
737   \ifx\titlepage@restore\relax\else\clearpage\titlepage@restore\fi
738 \end{report}
739 \end{article}
740 \end{report}
741 \setcounter{footnote}{0}
742 \expandafter\ifnum \csname scr@v03.12\endcsname>\scr@compatibility\relax
743   \let\thanks\relax
744   \let\maketitle\relax
745   \let\@maketitle\relax
746   \global\let\@thanks\@empty
747   \global\let\@author\@empty
748   \global\let\@date\@empty
749   \global\let\@title\@empty
750   \global\let\@subtitle\@empty
751   \global\let\@extratitle\@empty
752   \global\let\@frontispiece\@empty
753   \global\let\@titlehead\@empty
754   \global\let\@subject\@empty
755   \global\let\@publishers\@empty
756   \global\let\@uppertitleback\@empty
757   \global\let\@lowertitleback\@empty
758   \global\let\@dedication\@empty
759   \global\let\@matrikelnr\@empty
760   \global\let\@professor\@empty
761   \global\let\author\relax
762   \global\let\title\relax
763   \global\let\extratitle\relax
764   \global\let\titlehead\relax
765   \global\let\subject\relax
766   \global\let\publishers\relax
767   \global\let\uppertitleback\relax
768   \global\let\lowertitleback\relax
769   \global\let\dedication\relax
770   \global\let\date\relax
771   \global\let\matrikelnr\relax
772   \global\let\professor\relax
773 \fi
774 \global\let\and\relax

```

```

775 }
776 </article | report>

```

5.6 Localisation

The following section contains language specific definitions of text used by the classes.

```

\professorname Define the commands content for the different supported languages.
\firstexaminername 777 \newcaptionname{english}\professorname{Professor}
\secondexaminername 778 \newcaptionname{german,ngerman}\professorname{Professor(in)/Lehrbeauftragte(r)}
779 \newcaptionname{english}\firstexaminername{First examiner}
780 \newcaptionname{german,ngerman}\firstexaminername{Erstpr{\u}fer(in)}
781 \newcaptionname{english}\secondexaminername{Second examiner}
782 \newcaptionname{german,ngerman}\secondexaminername{Zweitpr{\u}fer(in)}

\decofauthname Define the german translations for the command.
783 \newcaptionname{english}\decofauthname{Declaration of Authorship}
784 \newcaptionname{german,ngerman}\decofauthname{Selbstst{\a}ndigkeitserkl{\a}rung}

\decofauthtext Define the german translations for the command.
785 \newcaptionname{english}\decofauthtext{%
786 \ifsingleauthor{I}{We} hereby certify that the work \ifsingleauthor{I}{we}
787 \ifsingleauthor{am}{are} submitting is entirely of \ifsingleauthor{my}{our}
788 own making except where otherwise indicated. \ifsingleauthor{I}{We}
789 \ifsingleauthor{am}{are} aware of regulations concerning plagiarism,
790 including disciplinary actions that may result from it. Any use of the
791 works of any other author, in any form, is properly acknowledged at
792 their point of use.
793 }
794 \newcaptionname{german,ngerman}\decofauthtext{%
795 Hiermit best{\a}tige\ifsingleauthor{}{n} \ifsingleauthor{ich}{wir},
796 dass die folgende Arbeit eigenst{\a}ndig von \ifsingleauthor{mir}{uns}
797 allein erstellt und unter Ber{\u}cksichtigung der zur Verf{\u}gung
798 gestellten Aufgabenstellung sowie dem Arbeitsmaterial unter Angabe aller
799 verwendeten Quellen erarbeitet wurde. Die Regelungen und Konsequenzen
800 eines Plagiats, inklusive disziplinarischer Ma{\ss}nahmen, sind
801 \ifsingleauthor{mir}{uns} bewusst. Insbesondere wurden alle Zitate und
802 gedanklichen {\u}bernahmen als solche kenntlich gemacht.
803 }

\keywordsname Define the german translations for the command.
804 \newcaptionname{english}\keywordsname{Keywords}
805 \newcaptionname{german,ngerman}\keywordsname{Schl{\u}sselw{\o}rter}

```