

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

Jan Wille[†]

Created 14.06.2024

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	10
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	14
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	22

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 forming basis that should work for pretty much everything a student 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 emssage 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 availabel 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 contentes.</td></tr> <tr> <td><code>notoc</code></td><td>The abstract will <i>not</i> be listed in the table of contentes.</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 contentes.	<code>notoc</code>	The abstract will <i>not</i> be listed in the table of contentes.		
<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 contentes.										
<code>notoc</code>	The abstract will <i>not</i> be listed in the table of contentes.										
<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 contentes will list itself.</td></tr> <tr> <td><code>notoc</code></td><td>The table of contentes will <i>not</i> list itself.</td></tr> <tr> <td><code>abstract</code></td><td>The abstract will be listed in the table of contentes.</td></tr> <tr> <td><code>noabstract</code></td><td>The abstract will <i>not</i> be listed in the table of contentes.</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 contentes will list itself.	<code>notoc</code>	The table of contentes will <i>not</i> list itself.	<code>abstract</code>	The abstract will be listed in the table of contentes.	<code>noabstract</code>	The abstract will <i>not</i> be listed in the table of contentes.	<code>...</code>	see KOMA-Script manual, Table 3.5 for more options
<code>totoc</code>	The table of contentes will list itself.										
<code>notoc</code>	The table of contentes will <i>not</i> list itself.										
<code>abstract</code>	The abstract will be listed in the table of contentes.										
<code>noabstract</code>	The abstract will <i>not</i> be listed in the table of contentes.										
<code>...</code>	see KOMA-Script manual, Table 3.5 for more options										

Often relevant KOMA-Script options

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

<code>fontsize</code>	The <code>fontsize=<size></code> options takes a size in pt. It is usually in therange 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 doublesided printing. The header and footer will take this into account and binding-correction will be applide along the inner edge.
<code>BCOR</code>	The <code>BCOR=<dim></code> option allows you to define a custom binding-correction. Any valide length can be put here, but to 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 controlls how much whicspace is added arround you standalone document. This usually looks better which is why the default is 0.25 cm, but you can supress it by passing 0 cm.
<code>multi</code>	The <code>multi=<opt></code> option defines which enviroment make up a page. It can be passe 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 classname. This is mostly useful so internals can reuse the classname, 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 her.

- `\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 Allways 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@chap{\addchap*}

```

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

```

110 \DefineFamilyKey{HsH}{abstract}{%
111 \begin{group}
112 \FamilySetNumerical{HsH}{abstract}{@tempa}{%
113 {keywords}{0},%
114 {nokeywords}{1},%
115 {totoc}{2},{toc}{2},%
116 {notoc}{3},{nottotoc}{3},%
117 }{#1}
118 \ifx\FamilyKeyState\FamilyKeyStateProcessed
119 \aftergroup\FamilyKeyStateProcessed
120 \ifcase\@tempa% 0
121 \endgroup
122 \@HsH@option@abstract@show@keywordstrue
123 \or% 1
124 \endgroup
125 \@HsH@option@abstract@show@keywordsfalse
126 \or% 2
127 \endgroup
128 \def\HsH@abstract@chap{\addchap}
129 \or% 3
130 \endgroup
131 \def\HsH@abstract@chap{\addchap*}
132 \else% should never be
133 \endgroup
134 \fi
135 \else
136 \endgroup
137 \fi
138 }

```

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

```

139 \DefineFamilyKey{HsH}{toc}{%
140 \begin{group}
141 \FamilySetNumerical{HsH}{toc}{@tempa}{%
142 {totoc}{0},{toc}{0},%
143 {notoc}{1},{nottotoc}{1},%
144 {abstract}{2},%
145 {noabstract}{3},%
146 }{#1}
147 \ifx\FamilyKeyState\FamilyKeyStateProcessed
148 \aftergroup\FamilyKeyStateProcessed
149 \ifcase\@tempa% 0
150 \endgroup

```



```

151 \AtEndOfClass{\setuptoc{toc}{totoc}}
152 \or% 1
153 \endgroup
154 \AtEndOfClass{\unsettoc{toc}{totoc}}
155 \or% 2
156 \endgroup
157 \FamilyExecuteOptions{HsH}{abstract=totoc}
158 \or% 3
159 \endgroup
160 \FamilyExecuteOptions{HsH}{abstract=nottotoc}
161 \else% should never be
162 \endgroup
163 \fi
164 \else
165 \endgroup
166 \HsH@Options@PassToParrent{toc=#1}
167 \FamilyKeyStateProcessed
168 \fi
169 }

```

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

```

170 \DefineFamilyKey{HsH}{twoside}[semi]{%
171 \HsH@Options@PassToParrent{twoside=#1,BCOR=1cm}
172 \FamilyKeyStateProcessed
173 }
174 </!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.

```
175 \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 times, but only the last set value will be passed on to babel.

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

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

```

german 177 \HsH@Options@DeclareAlias{english}{language=english}
ngerman 178 \HsH@Options@DeclareAlias{german}{language=ngerman}
179 \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 and initialised manually.

```
180 \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`.

```

181 \FamilyNumericalKey{HsH}{faculty}{\HsH@opt@faculty}{%
182 {0}{1}, {none}{1}, {false}{1}, {off}{1},%
183 {1}{2}, {f1}{2},%
184 {2}{3}, {f2}{3},%
185 {3}{4}, {f3}{4},%
186 {4}{5}, {f4}{5},%
187 {5}{6}, {f5}{6},%
188 }

```

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

```

f2 189 \HsH@Options@DeclareAlias{f1}{faculty=f1}
f3 190 \HsH@Options@DeclareAlias{f2}{faculty=f2}
f4 191 \HsH@Options@DeclareAlias{f3}{faculty=f3}
f5 192 \HsH@Options@DeclareAlias{f4}{faculty=f4}
193 \HsH@Options@DeclareAlias{f5}{faculty=f5}

```

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

```
194 \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 should create an annoying error just for this.

```
195 <*standalone>
196 \DefineFamilyKey{HsH}{fontsize}{%
197   \ClassInfoNoLine{\HsHClassName}{The 'fontsize' option is only a mock, its has not effect}
198   \FamilyKeyStateProcessed
199 }
200 </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 executed for every unknown key-value option. Unknown bare options are handled by the `\DeclareOption*` command and will be passed there.

```
201 \DefineFamilyKey{HsH}{@else@}{
202   \HsH@Options@PassToParrent{#1}
203   \FamilyKeyStateProcessed
204 }
205 \DeclareOption*{
206   \HsH@Options@PassToParrent{\CurrentOption}
207 }
```

5.2.2 Default options

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

```
208 \FamilyExecuteOptions{HsH}{%
209   fontfamily=sans-serif,
210 <!*standalone>
211   fontsize=11pt,
212   language=ngerman,
213 <article>   parskip=never+,
214 <report>   parskip=half+,
215   linespacing=single,
216   headheight=2.15\baselineskip,
217 </!standalone>
218 <*article | report>
219   toc=listof,
220   toc=bibliography,
221   abstract=keywords,
222 </article | report>
223   faculty=none,
224 <*standalone>
225   margin=0.25cm,
226   multi=tikzpicture,
227   multi=circuitikz,
228 </standalone>
229 }
```

Now we can process the options for the `HsH` family.

```
230 \FamilyProcessOptions{HsH}\relax
```

5.2.3 Loading the parent class

```
231 \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 characterset.

```
232 \RequirePackage[T1]{fontenc}
```

```
233 \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.

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

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

```
235 \RequirePackage[autostyle=true]{csquotes}
```

```
236 \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 standart.

```
237 \iflanguage{ngerman}{\RequirePackage{ziffer}}{}
```

5.3.2 Generally usefull packages

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

```
238 \RequirePackage[hidelinks]{hyperref} % must load before 'bookmarks'
```

```
239 \RequirePackage{bookmark}
```

```
240 \ifstandalone
```

```
241 \AtBeginDocument{
```

```
242   \hypersetup{
```

```
243     pdfinfo={
```

```
244       Title={\@title},
```

```
245       Author={\@author},
```

```
246       Subject={\@subject},
```

```
247       Keywords={\@keywords}
```

```
248     }
```

```
249   }
```

```
250 }
```

```
251 \ifstandalone
```

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.

```
252 \if@todos
```

```
253   \PassOptionsToPackage{
```

```
254     textsize=small,
```

```
255     figwidth=.6\textwidth
```

```
256   }{todonotes}
```

```
257   \RequirePackage{todonotes}
```

```
258 \else
```

```
259   \newcommand{\listoftodos}[1]{} 
```

```
260   \newcommand{\todo}[2] [] {} 
```

```
261   \newcommand{\missingfigure}[2] [] {} 
```

```
262 \fi
```

```

263 \RequirePackage[hypcap=true]{caption}
264 \RequirePackage{graphicx}
265 \RequirePackage{amsmath,amssymb,amsfonts}
266 \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.

```

267 (*article | report)
268 \PassOptionsToPackage{backend=biber}{biblatex}
269 \AtBeginDocument{
270   \makeatletter
271   \@ifpackageloaded{biblatex}{
272     \renewcommand*{\mkbibacro}[1]{\MakeUppercase{#1}}
273   }{}%
274   \makeatother
275 }
276 \end{document}

```

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

```

277 \PassOptionsToPackage{free-standing-units}{siunitx}
278 \AtBeginDocument{
279   \makeatletter
280   \@ifpackageloaded{siunitx}{
281     \iflanguage{ngerman}{
282       \sisetup{output-decimal-marker={,}}
283     }{}
284   }{}
285   \makeatother
286 }

```

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

```

287 \RequirePackage{scrhack}
288 \AtBeginDocument{
289   \makeatletter
290   \@ifpackageloaded{listings}{
291     \RequirePackage{lstautogobble}\lstset{autogobble=true}
292     \iflanguage{ngerman}{
293       \lstset{literate={Ö}{\ "0}}1{Ä}{\ "A}}1{Ü}{\ "U}}1{ß}{\ "ss}}1{ü}{\ "u}}1{ä}{\ "a}}1{ö}{\ "o}}1{}
294     }{}
295   }{}
296   \makeatother
297 }

```

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

```

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

```

For other packages we provide the settings more as a recomendation 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.

```

299 \AtBeginDocument{
300   \makeatletter
301   \InputIfFileExists{HsH-classes.cfg}{
302     \ClassInfo{\HsHClassName}{Local config file HsH-classes.cfg used.}
303   }{
304     \ClassInfo{\HsHClassName}{No HsH-classes.cfg!! I hope you configered it yourself.}
305   }

```

```

306 \makeatother
307 }

```

5.4 Custom commands

5.4.1 Document separation commands

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

```

308 \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 different.

```

309 \newif\if@mainmatter\@mainmattertrue

```

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

`\frontmatter` The page numbering is set to capital roman numerals.

```

310 \newcommand{\frontmatter}{
311   \if@twoside\cleardoubleoddpage\else\clearpage\fi
312   \@mainmatterfalse\pagenumbering{Roman}
313 }

```

`\mainmatter` The page numbering is set to arabic numerals.

```

314 \newcommand{\mainmatter}{
315   \if@twoside\cleardoubleoddpage\else\clearpage\fi
316   \@mainmattertrue\pagenumbering{arabic}
317 }

```

`\backmatter` The page numbering is set to arabic numerals.

```

318 \newcommand{\backmatter}{
319   \if@openright\cleardoubleoddpage\else\clearpage\fi
320   \@mainmatterfalse\pagenumbering{roman}
321 }

```

```

322 \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 display 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.

```

323 \newcommand{\HsHlogoPath}{HSH-Logo.pdf}

```

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

```

324 \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 5 cm.

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

```

325 \newcommand{\includeHsHlogohere}[1][5cm]{
326   \IfFileExists{\HsHlogoPath}{
327     \includegraphics[width=#1,page=\HsHlogoPage]{\HsHlogoPath}
328   }{
329     \ClassError{\HsHClassName}{\HsHlogoPath\space not found!}{
330       The HsH Logo is necessary for the titlepage! Try putting it next to your source file or
331     }
332   }
333 }

```

5.4.3 Title matters

334 \langle *article | report \rangle

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.

```
335 \renewcommand{\title}[2][]{
336   \gdef\@title{#2}
337   \gdef\@shorttitle{#1}
338 }
```

`\@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.

```
339 \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.

```
340 \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.

```
341 \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.

```
342 \let\@author\@empty
```

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

```
343 \newcommand{\matrikelnr}[1]{\gdef@with@and@as@comma\@matrikelnr{#1}}
344 \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.

```
\firstexaminer 345 \newcommand{\professor}[1]{\gdef\@professor{#1}}
\secondexaminer \@professor 346 \def\@professor{\@empty}
\@firstexaminer 347 \newcommand{\firstexaminer}[1]{\gdef\@firstexaminer{#1}}
\@secondexaminer 348 \def\@firstexaminer{\@empty}
349 \newcommand{\secondexaminer}[1]{\gdef\@secondexaminer{#1}}
350 \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.

```
351 \newcommand{\keywords}[1]{\gdef\@keywords{#1}}
352 \def\@keywords{\@empty}
```

353 \langle /article | report \rangle

5.4.4 Commands for document writing

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

354 \langle !standalone \rangle

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

```
355 \RequirePackage{pgffor}
```

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

```
356 \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.

```

357 \if#1b
358   \vspace*{\fill}
359   \hrule
360 \else\if#1t
361   \vspace*{2em}
362 \else
363   \ClassError{\HsHClassName}{Wrong Parameter for ‘\declarationofauthorship’}{
364     ‘\string\declarationofauthorship’ only accepts ‘t’ and ‘b’ as parameters.
365   }
366 \fi\fi

```

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

```

367 \vskip 3em
368 {\centering\bfseries\usekomafont{section}{\decofauthname}\par}
369 \vskip 3em
370 \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.

```

371 \begingroup
372   \renewcommand{\thanks}{\sbox0}
373   \raggedleft
374   \foreach \tmp@ in \@author {
375     \if\tmp@empty\else
376       \hskip 1em \parbox{4cm}{
377         \vskip 4em
378         \hrule\vskip 4pt
379         \raggedleft\footnotesize\tmp@
380       }%
381     \fi
382   }\par
383 \endgroup
384 }

```

\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.

```

385 \newif\if@singleauthor
386 \AtBeginDocument{
387   \begingroup
388     \newcount\count@
389     \count@=\z@
390     \@for\tmp@:=\@author\do{\advance\count@\@ne}
391     \ifnum\count@>\@ne\global\@singleauthorfalse\else\global\@singleauthortrue\fi
392   \endgroup
393 }

```

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

```

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

```

\declarationAuthorship

```

395 \def\declarationAuthorship{%
396   \ClassWarning{\HsHClassName}{%
397     Command \string\declarationAuthorship\space is deprecate.\MessageBreak
398     Replace it with \string\declarationofauthorship.
399   }%
400   \declarationofauthorship%
401 }
402 </!stadnalone>

```

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.

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

Configuring mathmode-indices

The only hard requirements for documents writing on Hochschule Hannover is, that the indices in mathematic formulas must be typeset 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 modifiable.

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

```
404 \def\@subinrm#1{\sb{\mathrm{#1}}}  
405 {\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.

```
406 \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.

```
407 \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

```
408 \RequirePackage{lmodern}
```

5.5.2 Page layout

```
409 \<!*standalone>  
410 \areaset[current]{0.75\paperwidth}{0.8\paperheight}  
411 \if@todos  
412   \addtolength\paperwidth{5cm}  
413   \addtolength\marginparwidth{5cm}  
414 \fi  
415 \</!*standalone>
```

5.5.3 Styling L^AT_EX default constucts

Floats

```
416 \<!*standalone>
```

Floats should alsway prefere the *here* placement, than the *top* of the following page.

```
417 \renewcommand{\fps@figure}{h!t}  
418 \renewcommand{\fps@table}{h!t}
```

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

```
419 \g@addto@macro\floatboxreset\centering  
420 \setcapwidth{0.8\textwidth}
```

The names of floating enviroments are redefined to show abreviations only.

```
421 \defcaptionname{english}\figurename{Fig.}  
422 \defcaptionname{german,ngerman}\figurename{Abb.}  
423 \defcaptionname{english}\tablename{Tab.}  
424 \defcaptionname{german,ngerman}\tablename{Tab.}
```


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

```
425 \AtBeginDocument{
426   \makeatletter
427   \@ifpackageloaded{subfigure}{
428     \let\subfigureautorefname\figureautorefname
429   }{}%
430   \makeatother
431 }
432 \</!standalone>
```

Lists

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

```
433 \renewcommand{\labelitemi}{\raisebox{.3ex}{\scalebox{0.7}{\bullet$}}}
434 \renewcommand{\labelitemii}{\raisebox{.3ex}{\scalebox{0.7}{\circ$}}}
435 \renewcommand{\labelitemiii}{\raisebox{.1ex}{-}}
436 \renewcommand{\labelitemiv}{\raisebox{-.1ex}{\scalebox{1.3}{\cdot$}}}
```

Abstract

We define some custom behavior for the abstract.

```
437 \renewenvironment{abstract}{
438   \quotation
439   \setparsizes{\z@}{\z@}{.25\linewidth \@plus 1fil}\selectfont
440   \HsH@abstract@chap{\abstractname}
441 }{%
442   \ifx\@keywords\@empty\else\if@HsH@option@abstract@show@keywords
443     \par\bigskip
444     \noindent\textbf{\keywordsname}\hskip 2em\@keywords
445   \fi\fi\par
446   \endquotation
447 }
```

Misc

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

```
448 \upsubscripts
```

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

```
449 \raggedbottom
```

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

```
450 \let\layercontentsmeasure\relax
```

5.5.4 Header and footer

```
451 \<*article | report>
```

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

```
452 \RequirePackage{scrlayer-scrpage}
453 \FamilyOptions{KOMA}{headsepline,singlespacing=true}
```

First we define the new pagestyle HsHheadings.

```
454 \newpagestyle{HsHheadings}{
455   {
456     \parbox[b]{\sls@headwidth}{
457       \LaTeXraggedright
458       \ifx\@shorttitle\@empty\@title\else\@shorttitle\fi
459     }%
460   }
461   {
462     \parbox[b]{\sls@headwidth}{
463       \LaTeXraggedleft
464       \leftmark
465     }
466   }
467 }
```

```

465     }%
466   }
467   {
468     \parbox[b]{.45\sls@headwidth}{
469       \LaTeXraggedright
470       \ifx\@shorttitle\@empty\@title\else\@shorttitle\fi
471     }%
472     \hfill
473     \parbox[b]{.45\sls@headwidth}{
474       \LaTeXraggedleft
475       \headmark
476     }%
477   }
478   (\textwidth,.1pt)
479 }{
480   {\pagemark}
481   {\hfill\pagemark}
482   {\hfill\pagemark}
483 }

```

Then all generic settings are applied:

```

484 \clearpairofpagestyles
485 \ofoot*{\pagemark}
486 \pagestyle{HsHheadings}
487 \article\automark{section}
488 \report\automark{chapter}
489 \report\renewcommand*{\chapterpagestyle}{HsHheadings}
490 \</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.

```

491 \<article | report>
492 \newtoks\@tabtoks
493 \newcommand\addtabtoks[1]{\global\@tabtoks\expandafter{\the\@tabtoks#1}}
494 \newcommand\eadddtabtoks[1]{\edef\mytmp{#1}\expandafter\addtabtoks\expandafter{\mytmp}}
495 %%\newcommand*\resettabtoks{\global\@tabtoks{}}
496 \newcommand*\printtabtoks{\the\@tabtoks}
497 \addtokomafont{publishers}{\normalsize}
498 \g@addto@macro\tITLEpage{\singlespacing}
499 %
500 \article\renewcommand\maketitle[1][c]{
501 \report\renewcommand\maketitle[1][l]{
502   \expandafter\ifnum \csname scr@v@3.12\endcsname>\scr@compatibility\relax
503   \else
504     \def\and{%
505       \end{tabular}
506       \hskip 1em \@plus.17fil
507       \begin{tabular}[t]{c}%
508     }
509     \fi
510 \<article>
511   \par
512   \ifx\@uppertitleback\@empty\else
513     \ClassWarning{\KOMAClassName}{%
514       non empty \string\uppertitleback\space ignored
515       by \string\maketitle\MessageBreak
516       in ‘titlepage=false’ mode%
517     }
518   \fi
519   \ifx\@lowertitleback\@empty\else
520     \ClassWarning{\KOMAClassName}{%
521       non empty \string\lowertitleback\space ignored
522       by \string\maketitle\MessageBreak

```

```

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

```

```

586      \noindent\@extratitle
587      \ifx\@frontispiece\@empty
588      \else
589      \next@tpage
590      \noindent\@frontispiece
591      \fi
592      \next@tdpage
593 \</report>
594 \fi
595 \<article>
596 \ifx\@frontispiece\@empty
597 \ifx\@extratitle\@empty\else\next@tdpage\fi
598 \else
599 \next@tpage
600 \makefrontispiece
601 \next@tdpage
602 \fi
603 \global\@topnum=\z@
604 \</article>
605 \setparsizes{\z@}{\z@}{\z@\@plus 1fil}\par@updaterelative
606 \vspace*{1cm}
607 \begin{minipage}[t]{\textwidth}%
608 \ifx\@titlehead\@empty \else
609 \usekomafont{titlehead}{\@titlehead}%
610 \fi
611 \hfill
612 % image with referencepoint in lower left corner:
613 \raisebox{0pt}[\ht\strutbox][\dp\strutbox]{\includeHsHlogohere}
614 \end{minipage}
615 \raisebox{10pt}{\rule{\textwidth}{0.5pt}}
616 \null
617 \<article> \vskip 2em
618 \<report> \vfill
619 \begin{group
620 \if\@param c\centering\fi
621 \if\@param r\raggedleft\fi
622 \ifx\@subject\@empty\else
623 {\usekomafont{subject}{\@subject\par}}
624 \<article> \vskip 1.5em
625 \<report> \vskip 3em
626 \fi
627 {\usekomafont{title}{\huge\@title\par}}
628 \<article> \vskip .5em
629 \<report> \vskip 1em
630 {\ifx\@subtitle\@empty\else\usekomafont{subtitle}\@subtitle\par\fi}
631 \<article|report> \vskip 4em
632 {\ifx\@matrikelnr\@empty
633 \if\@author\@empty\else\usekomafont{author}{
634 \parbox{\dimexpr\linewidth}{
635 \if\@param c\centering\fi
636 \if\@param r\raggedleft\fi
637 \@author
638 }
639 }\fi
640 \else
641 \if\@author\@empty\else
642 % sneaky comma needed after \@matrikelnr to deal with single item lists
643 \foreach \x [count=\i,evaluate=\i as \y using {\@matrikelnr,\i-1}] in \@author {
644 \usekomafont{author}{
645 \def\arraystretch{1.2}
646 \if\@param l\begin{tabular}{@{}l l}\fi
647 \if\@param c\begin{tabular}{l l}\fi
648 \if\@param r\begin{tabular}{r r@{}}\fi

```

```

649          \printtabtoks
650        \end{tabular}
651      }%
652    \fi
653  \fi}
654 <article>      \vskip 1em
655 <report>       \vskip 1.5em
656      {\usekomafont{date}{\@date\par}}
657 <article>      \vskip 1em
658 <report>       \vskip \z@ \@plus3fill
659      \usekomafont{publishers}{
660        \def\arraystretch{1.2}
661        \if\@param l\begin{tabular}{@{}l l}\fi
662        \if\@param c\begin{tabular}{l l}\fi
663        \if\@param r\begin{tabular}{r r@{}}\fi
664        \if\@professor\@empty\else\textbf{\professorname:}&\@professor\\\fi
665        \if\@firstexaminer\@empty\else\textbf{\firstexaminername:}&\@firstexaminer\\\fi
666        \if\@secondexaminer\@empty\else\textbf{\secondexaminername:}&\@secondexaminer\\\fi
667      \end{tabular}
668    }
669 <*article>
670      \ifx\@dedication\@empty\else
671        \vskip 2em
672        {\usekomafont{dedication}{\@dedication \par}}%
673      \fi
674 </article>
675      \par
676    \endgroup
677 <article>      \vskip 2em
678 <report>       \vskip 3em
679 <article>      \ifx\titlepagestyle\@empty\else\thispagestyle{\titlepagestyle}\fi
680      \@thanks\global\let\@thanks\@empty
681 <*report>
682      \vfill\null
683      \if@twoside
684        \@tempswatrue
685        \expandafter\ifnum \@nameuse{scr@v@3.12}>\scr@compatibility\relax
686        \else
687          \ifx\@uppertitleback\@empty\ifx\@lowertitleback\@empty
688            \@tempswafalse
689          \fi\fi
690        \fi
691        \if@tempswa
692          \next@tpage
693          \begin{minipage}[t]{\textwidth}
694            \@uppertitleback
695          \end{minipage}\par
696          \vfill
697          \begin{minipage}[b]{\textwidth}
698            \@lowertitleback
699          \end{minipage}\par
700          \@thanks\global\let\@thanks\@empty
701        \fi
702      \else
703        \ifx\@uppertitleback\@empty\else
704          \ClassWarning{\KOMAClassName}{%
705            non empty \string\uppertitleback\space ignored
706            by \string\maketitle\MessageBreak
707            in ‘twoside=false’ mode%
708          }
709        \fi
710        \ifx\@lowertitleback\@empty\else
711          \ClassWarning{\KOMAClassName}{%

```

```

712         non empty \string\lowertitleback\space ignored
713         by \string\maketitle\MessageBreak
714         in 'twoside=false' mode%
715     }
716     \fi
717 \fi
718 \ifx\@dedication\@empty
719 \else
720     \next@tdpage\null\vfill
721     {\centering\usekomafont{dedication}{\@dedication \par}}%
722     \vskip \z@ \@plus3fill
723     \@thanks\global\let\@thanks\@empty
724     \cleardoubleemptypage
725 \fi
726 \ifx\titlepage@restore\relax\else\clearpage\titlepage@restore\fi
727 \</report>
728 \<article> \endgroup
729 \<report> \end{titlepage}
730 \setcounter{footnote}{0}
731 \expandafter\ifnum \csname scr@v@3.12\endcsname>\scr@compatibility\relax
732 \let\thanks\relax
733 \let\maketitle\relax
734 \let\@maketitle\relax
735 \global\let\@thanks\@empty
736 \global\let\@author\@empty
737 \global\let\@date\@empty
738 \global\let\@title\@empty
739 \global\let\@subtitle\@empty
740 \global\let\@extratitle\@empty
741 \global\let\@frontispiece\@empty
742 \global\let\@titlehead\@empty
743 \global\let\@subject\@empty
744 \global\let\@publishers\@empty
745 \global\let\@uppertitleback\@empty
746 \global\let\@lowertitleback\@empty
747 \global\let\@dedication\@empty
748 \global\let\@matrikelnr\@empty
749 \global\let\@professor\@empty
750 \global\let\author\relax
751 \global\let\title\relax
752 \global\let\extratitle\relax
753 \global\let\titlehead\relax
754 \global\let\subject\relax
755 \global\let\publishers\relax
756 \global\let\uppertitleback\relax
757 \global\let\lowertitleback\relax
758 \global\let\dedication\relax
759 \global\let\date\relax
760 \global\let\matrikelnr\relax
761 \global\let\professor\relax
762 \fi
763 \global\let\and\relax
764 }
765 \</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 766 \newcaptionname{english}\professorname{Professor}
\secondexaminername 767 \newcaptionname{german,ngerman}\professorname{Professor(in)/Lehrbeauftragte(r)}
768 \newcaptionname{english}\firstexaminername{First examiner}

```

```

769 \newcaptionname{german,ngerman}\firstexaminername{Erstpr{"u}fer(in)}
770 \newcaptionname{english}\secondexaminername{Second examiner}
771 \newcaptionname{german,ngerman}\secondexaminername{Zweitpr{"u}fer(in)}

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

\decofauthtext Define the german translations for the command.
774 \newcaptionname{english}\decofauthtext{%
775 \ifsingleauthor{I}{We} hereby certify that the work \ifsingleauthor{I}{we}
776 \ifsingleauthor{am}{are} submitting is entirely of \ifsingleauthor{my}{our}
777 own making except where otherwise indicated. \ifsingleauthor{I}{We}
778 \ifsingleauthor{am}{are} aware of regulations concerning plagiarism,
779 including disciplinary actions that may result from it. Any use of the
780 works of any other author, in any form, is properly acknowledged at
781 their point of use.
782 }
783 \newcaptionname{german,ngerman}\decofauthtext{%
784 Hiermit best{"a}tuge\ifsingleauthor{}{n} \ifsingleauthor{ich}{wir},
785 dass die folgende Arbeit eigenst{"a}ndig von \ifsingleauthor{mir}{uns}
786 allein erstellt und unter Ber{"u}cksichtigung der zur Verf{"u}gung
787 gestellten Aufgabenstellung sowie dem Arbeitsmaterial unter Angabe aller
788 verwendeten Quellen erarbeitet wurde. Die Regelungen und Konsequenzen
789 eines Plagiats, inklusive disziplinarischer Ma{\ss}nahmen, sind
790 \ifsingleauthor{mir}{uns} bewusst. Insbesondere wurden alle Zitate und
791 gedanklichen {"U}bernahmen als solche kenntlich gemacht.
792 }

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

```