

HsH-Classes — A set of L^AT_EX classes for use in Hochschule Hannover *

Jan Wille[†]

Printed 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	10
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 L ^A T _E X Commands that are modified	4	5.4.2 The Logo for Hochschule Hannover	13
4 Package loading	4	5.4.3 Title matters	14
4.1 Allways loaded packages	5	5.4.4 Commands for document writing	14
4.2 Conditionally loaded Packages	5	5.4.5 Micalanious commands	15
4.3 Pre-configured Packages	5	5.5 Document setup	16
5 Implementation	5	5.5.1 Fonts and text styling	16
5.1 Internal commands	5	5.5.2 Page layout	16
5.2 Option handling	5	5.5.3 Styling L ^A T _E X default constucts	16
5.2.1 Unknown options	10	5.5.4 Header and footer	17
		5.5.5 Titlepage	18
		5.6 Localisation	22

Abstract

The following documents a set of L^AT_EX 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 requiements, 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 3.02, last revised 2024-05-02.

[†]E-mail: mail@janiwille.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.

<code>\matrikelnr</code>	The <code>\matrikelnr{<nr[, ..>}</code> 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 <code>\author</code> .
<code>\professor</code>	You can pass any text to <code>\professor{<text>}</code> , it will be printed on the bottom of the titelpage.
<code>\keywords</code>	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

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

3.2 Commands for document writing

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

<code>\declarationofauthorship</code>	The command <code>\declarationofauthorship[<align>]</code> can be used to print a "declatation of authorship" in the current location, similar to how <code>tableofcontents</code> and <code>friends</code> 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 <code>t</code> for alignment at the top of the page and <code>b</code> for bottom alignment (<i>default</i>).
<code>\frontmatter</code>	The three commands <code>\frontmatter</code> , <code>\mainmatter</code> and <code>\backmatter</code> are provided for report classes. They separete document section and automatically set up <code>pagenumer</code> styles.
<code>\mainmatter</code>	<code>\frontmatter</code> set the pagenumers to capital roman numerals. This is usually required for everything before the first chapter.
<code>\backmatter</code>	<code>\mainmatter</code> sets the pagenumbering to "normal" arabic numbers. This is usually the style for the document content.
<code>\backmatter</code>	<code>\backmatter</code> can be used for appendixes 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 moddified to better fit this class. This is documented here.

<code>\title</code>	The <code>\title[<short-title>]{<title>}</code> 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.
<code>\maketitle</code>	The <code>\maketitle</code> command if L ^A T _E X default way to create a titlepage. We redefine it to produce a titlepage that matches the sytel typically used on the Hochschule Hannover. This incudes the logo beeing printed, depending on the <code>f1</code> to <code>f5</code> documentoptions. The command now also takes an options alignment-parameter: <code>\maketitle[<align>]</code> . You can pass <code>l</code> <code>c</code> <code>r</code> to get <i>left</i> , <i>center</i> or <i>right</i> alignment.

4 Package loading

The classes load some packges 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 quotation 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 `scrbase` package is loaded.

```
7 \RequirePackage{scrbase}
```

We also require some packages for some of the options, 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 option 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 convinience, 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 is a KOMA-Script option that gets 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 \begin{group}
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 sets 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\ifHsH@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   \begingroup
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   \begingroup
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, everyting 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 than 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 bable.

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

`english` For convinience, 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 an initalised 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 convinience, 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 allready default to `<true>` if called without and 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 <{*!standalone}
241 \AtBeginDocument{
242   \hypersetup{
243     pdfinfo={
244       Title={\@title},
245       Author={\@author},
246       Subject={\@subject},
247       Keywords={\@keywords}
248     }
249   }
250 }
251 </!standalone>
```

The todonotes package is greate for anotation, but extremly expensive on compiletime. So we load it only if the user requests it. Also its commands are stubed, so that they can be left in the sourcecode and jut 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{iterate={Ö}{\0}}{Ä}{\A}{Ü}{\U}{ß}{\ss}{ü}{\u}{ä}{\a}{ö}{\o}
294     }{}
295   }{}
296   \makeatother
297 }
```

The circuitikz needs to 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 recommendation of what is usefull.

As the user might want to change these, we outsource this to a separat 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 seperation commands

The following commands are only defined for book type classes by default. But they are also usefull 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 pagenummer 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 pagenummering is set to capital roman numerals.

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

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

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

`\backmatter` The pagenummering 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 documentoption 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 u
331     }
332   }
333 }
```

5.4.3 Title matters

334 (*article | report)

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

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

335 \let\@author\@empty

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

336 \renewcommand{\title}[2][]{}

337 \gdef\@title{#2}

338 \gdef\@shorttitle{#1}

339 }

`\@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.

340 \def\@shorttitle{\@empty}

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

`\@matrikelnr`

341 \newcommand{\matrikelnr}[1]{\gdef\@matrikelnr{#1}}

342 \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` 343 \newcommand{\professor}[1]{\gdef\@professor{#1}}

`\secondexaminer` 344 \def\@professor{\@empty}

`\@professor`

`\@firstexaminer` 345 \newcommand{\firstexaminer}[1]{\gdef\@firstexaminer{#1}}

`\@secondexaminer` 346 \def\@firstexaminer{\@empty}

347 \newcommand{\secondexaminer}[1]{\gdef\@secondexaminer{#1}}

348 \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.

`\@keywords`

349 \newcommand{\keywords}[1]{\gdef\@keywords{#1}}

350 \def\@keywords{\@empty}

351 (/article | report)

5.4.4 Commands for document writing

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

352 (*!standalone)

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

353 \RequirePackage{pgffor}

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

354 \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.

355 \if#1b

356 \vspace*{\fill}

357 \hrule

358 \else\if#1t

359 \vspace*{2em}

360 \else

361 \ClassError{\HsHClassName}{Wrong Parameter for '\declarationofauthorship'}{

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

363 }

364 \fi\fi

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

```
365 \vskip 3em
366 {\centering\bfseries\usekomafont{section}{\decofauthname}\par}
367 \vskip 3em
368 \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.

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

`\ifsingleauthor` 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.

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

`\declarationAuthorship`

```
392 \def\declarationAuthorship{%
393   \ClassWarning{HsHClassName}{%
394     Command \string\declarationAuthorship\space is deprecated.\MessageBreak
395     Replace it with \string\declarationofauthorship.
396   }%
397   \declarationofauthorship%
398 }
399 \!
```

5.4.5 Miscellaneous commands

Utility commands

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

```
400 \newcommand{\abs}[1]{\ensuremath{\left\vert\right.#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 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 then assign this to `_`. For that to work we need to change its catcode to make it modifiable.

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

```

401 \def\@subinrm#1{\sb{\mathrm{#1}}}
402 {\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.

```

403 \newcommand\upsubscripts{\catcode'\_ =12}

```

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

```

404 \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

```

405 \RequirePackage{lmodern}

```

5.5.2 Page layout

```

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

```

5.5.3 Styling L^AT_EX default constructs

Floats

```

413 \<!*standalone>

```

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

```

414 \renewcommand{\fps@figure}{h!t}
415 \renewcommand{\fps@table}{h!t}

```

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

```

416 \g@addto@macro\@floatboxreset\centering
417 \setcapwidth{0.8\textwidth}

```

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

```

418 \defcaptionname{english}\figurename{Fig.}
419 \defcaptionname{german,ngerman}\figurename{Abb.}
420 \defcaptionname{english}\tablename{Tab.}
421 \defcaptionname{german,ngerman}\tablename{Tab.}

```

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

```

422 \AtBeginDocument{
423   \makeatletter
424   \ifpackageloaded{subfigure}{
425     \let\subfigureautorefname\figureautorefname
426   }{}%
427   \makeatother
428 }
429 \</!standalone>

```

Lists

For unordert lists the markers are redefined to look a little nicer.

```

430 \renewcommand{\labelitemi}{\raisebox{.3ex}{\scalebox{0.7}{\bullet$}}}
431 \renewcommand{\labelitemii}{\raisebox{.3ex}{\scalebox{0.7}{\circ$}}}
432 \renewcommand{\labelitemiii}{\raisebox{.1ex}{-}}
433 \renewcommand{\labelitemiv}{\raisebox{-.1ex}{\scalebox{1.3}{\cdot$}}}

```


Abstract

We define some custom behavior for the abstract.

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

Misc

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

```
445 \upsubscripts
```

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

```
446 \raggedbottom
```

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

```
447 \let\layercontentsmeasure\relax
```

5.5.4 Header and footer

```
448 (*article | report)
```

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

```
449 \RequirePackage{scrlayer-scrpage}
450 \FamilyOptions{KOMA}{headsepline,singlespacing=true}
```

First we define the new pagestyle `HsHheadings`.

```
451 \newpagestyle{HsHheadings}{
452   {
453     \parbox[b]{\sls@headwidth}{
454       \LaTeXraggedright
455       \ifx\@shorttitle\@empty\@title\else\@shorttitle\fi
456     }%
457   }
458   {
459     \parbox[b]{\sls@headwidth}{
460       \LaTeXraggedleft
461       \leftmark
462     }%
463   }
464   {
465     \parbox[b]{.45\sls@headwidth}{
466       \LaTeXraggedright
467       \ifx\@shorttitle\@empty\@title\else\@shorttitle\fi
468     }%
469     \hfill
470     \parbox[b]{.45\sls@headwidth}{
471       \LaTeXraggedleft
472       \headmark
473     }%
474   }
475   (\textwidth,.1pt)
476 }{
477   {\pagemark}
478   {\hfill\pagemark}
479   {\hfill\pagemark}
480 }
```

Then all generic settings are applied:

```

481 \clearpairofpagestyles
482 \ofoot*{\pagemark}
483 \pagestyle{HsHheadings}
484 <article>\automark{section}
485 <report>\automark{chapter}
486 <report>\renewcommand*{\chapterpagestyle}{HsHheadings}
487 </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.

```

488 <*article|report>
489 \newtoks\@tabtoks
490 \newcommand\addtabtoks[1]{\global\@tabtoks\expandafter{\the\@tabtoks#1}}
491 \newcommand\eadddtabtoks[1]{\edef\mytmp{#1}\expandafter\addtabtoks\expandafter{\mytmp}}
492 %%\newcommand*\resettabtoks{\global\@tabtoks{}}
493 \newcommand*\printtabtoks{\the\@tabtoks}
494 \addtokomafont{publishers}{\normalsize}
495 \g@addto@macro\titlepage{\singlespacing}
496 %
497 <article>\renewcommand\maketitle[1][c]{
498 <report>\renewcommand\maketitle[1][l]{
499 \expandafter\ifnum \cscname scr@v03.12\endcscname>\scr@compatibility\relax
500 \else
501 \def\and{%
502 \end{tabular}
503 \hskip 1em \@plus.17fil
504 \begin{tabular}[t]{c}%
505 }
506 \fi
507 <*article>
508 \par
509 \ifx\@uppertitleback\@empty\else
510 \ClassWarning{\KOMAClassName}{%
511 non empty \string\uppertitleback\space ignored
512 by \string\maketitle\MessageBreak
513 in 'titlepage=false' mode%
514 }
515 \fi
516 \ifx\@lowertitleback\@empty\else
517 \ClassWarning{\KOMAClassName}{%
518 non empty \string\lowertitleback\space ignored
519 by \string\maketitle\MessageBreak
520 in 'titlepage=false' mode%
521 }
522 \fi
523 </article>
524 <report> \begin{titlepage}
525 <article> \begin{group
526 \let\@param#1
527 \ifx\@param\@empty
528 \ClassError{\myClassName}{\maketitle\space with empty option}{
529 \maketitle[] has been called (with an empty parameter), this doesn't work.
530 Use \maketitle instead.
531 }
532 \fi
533 <*report>
534 \if@titlepageiscoverpage
535 \edef\titlepage@restore{
536 \noexpand\endgroup
537 \noexpand\global\noexpand\@colht\the\@colht
538 \noexpand\global\noexpand\@colroom\the\@colroom

```

```

539     \noexpand\global\vsizethe\vsiz
540     \noexpand\global\noexpand\@titlepageiscoverpagefalse
541     \noexpand\let\noexpand\titlepage@restore\noexpand\relax
542 }
543 \begingroup
544 \topmargin=\dimexpr \coverpagetopmargin-1in\relax
545 \oddsidemargin=\dimexpr \coverpageleftmargin-1in\relax
546 \evensidemargin=\dimexpr \coverpageleftmargin-1in\relax
547 \textwidth=\dimexpr
548 \paperwidth-\coverpageleftmargin-\coverpagerightmargin\relax
549 \textheight=\dimexpr
550 \paperheight-\coverpagetopmargin-\coverpagebottommargin\relax
551 \headheight=0pt
552 \headsep=0pt
553 \footskip=\baselineskip
554 \@colht=\textheight
555 \@colroom=\textheight
556 \vsiz=\textheight
557 \columnwidth=\textwidth
558 \hsize=\columnwidth
559 \linewidth=\hsize
560 \else
561     \let\titlepage@restore\relax
562 \fi
563 \let\footnotesize\small
564 \let\footnoterule\relax
565 \let\footnote\thanks
566 \</report>
567 \<article> \let\titlepage@restore\relax
568     \renewcommand*\thefootnote{\@fnsymbol\c@footnote}%
569     \let\@oldmakefnmark\@makefnmark
570     \renewcommand*\@makefnmark{\rlap\@oldmakefnmark}%
571 \<article> \next@tdpage
572 \ifx\@extratitle\@empty
573 \<article> \ifx\@frontispiece\@empty\else \mbox{}\fi
574 \<report>
575     \ifx\@frontispiece\@empty\else
576         \if@twoside\mbox{}\fi
577         \noindent\@frontispiece\next@tdpage
578     \fi
579 \</report>
580 \else
581 \<article> \@makeextratitle
582 \<report>
583     \noindent\@extratitle
584     \ifx\@frontispiece\@empty
585     \else
586         \next@tdpage
587         \noindent\@frontispiece
588     \fi
589     \next@tdpage
590 \</report>
591 \fi
592 \<report>
593     \ifx\@frontispiece\@empty
594     \ifx\@extratitle\@empty\else\next@tdpage\fi
595     \else
596         \next@tdpage
597         \@makefrontispiece
598         \next@tdpage
599     \fi
600     \global\@topnum=\z@
601 \</article>

```

```

602 \setparsizes{\z@}{\z@}{\z@\@plus 1fil}\par@updaterelative
603 \vspace*{1cm}
604 \begin{minipage}[t]{\textwidth}%
605 \ifx\@titlehead\@empty \else
606 \usekomafont{titlehead}{\@titlehead}%
607 \fi
608 \hfill
609 % image with referencepoint in lower left corner:
610 \raisebox{0pt}[\ht\strutbox][\dp\strutbox]{\includeHsHlogohere}
611 \end{minipage}
612 \raisebox{10pt}{\rule{\textwidth}{0.5pt}}
613 \null
614 <article> \vskip 2em
615 <report> \vfill
616 \begin{group
617 \if\@param c\centering\fi
618 \if\@param r\raggedleft\fi
619 \ifx\@subject\@empty\else
620 {\usekomafont{subject}{\@subject\par}}
621 <article> \vskip 1.5em
622 <report> \vskip 3em
623 \fi
624 {\usekomafont{title}{\huge\@title\par}}
625 <article> \vskip .5em
626 <report> \vskip 1em
627 {\ifx\@subtitle\@empty\else\usekomafont{subtitle}\@subtitle\par\fi}
628 <article|report> \vskip 4em
629 {\ifx\@matrikelnr\@empty
630 \if\@author\@empty\else\usekomafont{author}{
631 \parbox{\dimexpr\linewidth}{
632 \if\@param c\centering\fi
633 \if\@param r\raggedleft\fi
634 \@author
635 }
636 }\fi
637 \else
638 \if\@author\@empty\else
639 % sneaky comma needed after \@matrikelnr to deal with single item lists
640 \foreach \x [count=\i,evaluate=\i as \y using {\@matrikelnr},{\i-1}] in \@author {
641 \usekomafont{author}{
642 \def\arraystretch{1.2}
643 \if\@param l\begin{tabular}{@{}l l}\printtabtoks\end{tabular}\fi
644 \if\@param c\begin{tabular}{l l}\printtabtoks\end{tabular}\fi
645 \if\@param r\begin{tabular}{r r@{}}\printtabtoks\end{tabular}\fi
646 }%
647 \fi
648 \fi}
649 <article> \vskip 1em
650 <report> \vskip 1.5em
651 {\usekomafont{date}{\@date\par}}
652 <article> \vskip 1em
653 <report> \vskip \z@ \@plus3fill
654 \usekomafont{publishers}{
655 \def\arraystretch{1.2}
656 \if\@param l\begin{tabular}{@{}l l}\fi
657 \if\@param c\begin{tabular}{l l}\fi
658 \if\@param r\begin{tabular}{r r@{}}\fi
659 \if\@professor\@empty\else\textbf{\@professorname:}&\@professor\\\fi
660 \if\@firstexaminer\@empty\else\textbf{\@firstexaminername:}&\@firstexaminer\\\fi
661 \if\@secondexaminer\@empty\else\textbf{\@secondexaminername:}&\@secondexaminer\\\fi
662 \end{tabular}
663 }
664 <*article>

```

```

665     \ifx\@dedication\@empty\else
666         \vskip 2em
667         {\usekomafont{dedication}{\@dedication \par}}%
668     \fi
669 \end{article}
670 \par
671 \endgroup
672 \article \vskip 2em
673 \report \vskip 3em
674 \article \ifx\titlepagestyle\@empty\else\thispagestyle{\titlepagestyle}\fi
675 \@thanks\global\let\@thanks\@empty
676 \*report
677 \vfill\null
678 \if@twoside
679     \@tempwatrue
680     \expandafter\ifnum \@nameuse{scr@v@3.12}>\scr@compatibility\relax
681     \else
682         \ifx\@uppertitleback\@empty\ifx\@lowertitleback\@empty
683             \@tempwafalse
684         \fi\fi
685     \fi
686     \if@tempswa
687         \next@tpage
688         \begin{minipage}[t]{\textwidth}
689             \@uppertitleback
690         \end{minipage}\par
691         \vfill
692         \begin{minipage}[b]{\textwidth}
693             \@lowertitleback
694         \end{minipage}\par
695         \@thanks\global\let\@thanks\@empty
696     \fi
697 \else
698     \ifx\@uppertitleback\@empty\else
699         \ClassWarning{\KOMAClassName}{%
700             non empty \string\uppertitleback\space ignored
701             by \string\maketitle\MessageBreak
702             in 'twoside=false' mode%
703         }
704     \fi
705     \ifx\@lowertitleback\@empty\else
706         \ClassWarning{\KOMAClassName}{%
707             non empty \string\lowertitleback\space ignored
708             by \string\maketitle\MessageBreak
709             in 'twoside=false' mode%
710         }
711     \fi
712 \fi
713 \ifx\@dedication\@empty
714 \else
715     \next@tdpage\null\vfill
716     {\centering\usekomafont{dedication}{\@dedication \par}}%
717     \vskip \z@ \@plus3fill
718     \@thanks\global\let\@thanks\@empty
719     \cleardoubleemptypage
720 \fi
721 \ifx\titlepage@restore\relax\else\clearpage\titlepage@restore\fi
722 \end{report}
723 \article \endgroup
724 \report \end{titlepage}
725 \setcounter{footnote}{0}
726 \expandafter\ifnum \csname scr@v@3.12\endcsname>\scr@compatibility\relax
727 \let\@thanks\relax

```

```

728 \let\maketitle\relax
729 \let\@maketitle\relax
730 \global\let\@thanks\@empty
731 \global\let\@author\@empty
732 \global\let\@date\@empty
733 \global\let\@title\@empty
734 \global\let\@subtitle\@empty
735 \global\let\@extratitle\@empty
736 \global\let\@frontispiece\@empty
737 \global\let\@titlehead\@empty
738 \global\let\@subject\@empty
739 \global\let\@publishers\@empty
740 \global\let\@uppertitleback\@empty
741 \global\let\@lowertitleback\@empty
742 \global\let\@dedication\@empty
743 \global\let\@matrikelnr\@empty
744 \global\let\@professor\@empty
745 \global\let\author\relax
746 \global\let\title\relax
747 \global\let\extratitle\relax
748 \global\let\titlehead\relax
749 \global\let\subject\relax
750 \global\let\publishers\relax
751 \global\let\uppertitleback\relax
752 \global\let\lowertitleback\relax
753 \global\let\dedication\relax
754 \global\let\date\relax
755 \global\let\matrikelnr\relax
756 \global\let\professor\relax
757 \fi
758 \global\let\and\relax
759 }
760 </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 761 \newcaptionname{english}\professorname{Professor}
\secondexaminername 762 \newcaptionname{german,ngerman}\professorname{Professor(in)/Lehrbeauftragte(r)}
763 \newcaptionname{english}\firstexaminername{First examiner}
764 \newcaptionname{german,ngerman}\firstexaminername{Erstpr{"u}fer(in)}
765 \newcaptionname{english}\secondexaminername{Second examiner}
766 \newcaptionname{german,ngerman}\secondexaminername{Zweitpr{"u}fer(in)}

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

\decofauthtext Define the german translations for the command.
769 \newcaptionname{english}\decofauthtext{%
770 \ifsingleauthor{I}\else{We}\fi\space hereby certify that the work
771 \ifsingleauthor{I}\else{we}\fi\space \ifsingleauthor am\else are\fi
772 submitting is entirely of \ifsingleauthor{my}\else{our}\fi own making
773 except where otherwise indicated. \ifsingleauthor{I}\else{We}\fi
774 \ifsingleauthor{am}\else{are}\fi\space aware of regulations concerning
775 plagiarism, including disciplinary actions that may result from it. Any
776 use of the works of any other author, in any form, is properly
777 acknowledged at their point of use.
778 }
779 \newcaptionname{german,ngerman}\decofauthtext{%
780 Hiermit best{"a}tige\ifsingleauthor\else{n}\fi

```

```

781 \ifsingleauthor{ich}\else{wir}\fi, dass die folgende Arbeit eigenst{"a}ndig
782 von \ifsingleauthor{mir}\else{uns}\fi\space allein erstellt und unter
783 Ber{"u}cksichtigung der zur Verf{"u}gung gestellten Aufgabenstellung sowie
784 dem Arbeitsmaterial unter Angabe aller verwendeten Quellen erarbeitet wurde.
785 Die Regelungen und Konsequenzen eines Plagiats, inklusive disziplinarischer
786 Ma{\ss}nahmen, sind \ifsingleauthor{mir}\else{uns}\fi\space bewusst.
787 Insbesondere wurden alle Zitate und gedanklichen {"U}bernahmen als
788 solche kenntlich gemacht.
789 }

```

\keywordsname Define the german translations for the command.

```

790 \newcaptionname{english}\keywordsname{Keywords}
791 \newcaptionname{german,ngerman}\keywordsname{Schl{"u}sselw{"o}rter}

```