TUDa-CI – Corporate Design for TU Darmstadt using LaTeX

Documentation for the TUDa-CI bundle

Marei Peischl*

v4.06, dated 2026-06-08

*Email: tuda-ci@peitex.de

TECHNISCHE UNIVERSITÄT DARMSTADT

Abstract

The TUDa-CI-Bundle provides a possibility to use the Corporate Design of TU Darmstadt with LaTeX. Therefore, it contains document classes as well as some helper packages and config files together with some templates for user documentation. Up to Version 4.00 the documentation was only included in the demo files. This document now sums up all features and includes references to other package documentations if required.

This process is not yet finished and will probably last some more time. So there will be more documentation as soon as possible and we try to gather everything within this document. There are possibilities to help with the documentation especially the translation of the descriptions already aailable in German. In case you want to help feel free to provide a pull-request via the GitHub repository or contact the author for other options of contributing.

Contents

1 Contents of the TUDa-CI Bundle

The TUDa-CI Bundle currently contains template files for the following document types:

Print publications generic document class which provides modes for more specific documents. The basic documentation applies to all types and can be found in chapter 3.

- `DEMO-TUDaPub.tex`: default document matching the CI guideline
- minimal Template for internal reports `DEMO-TUDaReport.tex` using type=intern (see description of type in section 3.1 on 7)
- (PhD) theses, using type=thesis. This also extends the functionality and adds some thesis specific mechanisms, see section 3.4 `DEMO-TUDaThesis.tex` and `DEMO-TUDaPhD.tex`

Scientific Posters based on the tcolorbox poster library, [9]

Presentation Slides beamer theme. This includes the old design of 2008 as well as the redesign of 2023, [8]

Announcement Posters for event or job announcements, [14]

Leaflets [11]

Exercise Sheets/Exams [13]

Letters [12]

The document classes use internal auxiliary packages to simplify the usage and reduce the maintenance effort. These are called by the elements they define: tudacolors, tudafonts, tudarules as well as the pgfplots color schemes defined by tuda-pgfplots.

Additionally, the setup is defined to support department specific configuration files. Officially we currently only support the setup of the department of mechanical engineering, see section 4.1. There may be unofficial custom setups within the departments. Those are not officially supported. In case you want to provide your own and add it to TUDa-CI, please have a look at chapter 4.

2 Getting Started

The TUDa-CI bundle is available via CTAN[10] and therefore no manual installation should be required. For TUDa members the HRZ also provides an on-premise Overleaf instance which also contains a release including all necessary files. In case you are a member of TU Darmstadt you can access the server via https://sharelatex.tu-darmstadt.de/.

2.1 Logo installation

Sadly the logo files may not be published together with this template. So it is necessary to either disable the loading or install the files. The easiest way to do so is by placing a copy of the required logos in your working directory. But this will be required for every document you want to use these templates for.

A better alternative would be a local installation of the logo file. This can be done by moving the file info your `TEXMFHOME` directory. On standard TEX Live Installations this would be called `texmf/` in your users home directory. Inside this one the logo files have to be placed according to the TDS (TEX Directory Structure), which means it has to be copied into `texmf/tex/latex/tuda-ci-logos`.

Alternatively there are scripts to be run inside a terminal to automatically detect the correct location. These can be found inside the `tools` directory of TUDa-CI's GitHub Repository (https://github.com/tudace/tuda_latex_templates/).

3 tudapub – generic class for articles and theses

The TUDa-CI bundle uses a generic document class to be more flexible with the layout adjustments. The setup is based on KOMA-Script and this section is listing the features. Afterwards there will be specific information on the thesis modes.

3.1 Usage and class options

The class is loaded using

```
\documentclass[⟨Options⟩]{tudapub}
```

TUDa-CI defines some additional options to be used. These are described below. The shown default values apply only for the document class itself. The template files may use different settings. An overview of these differences are shown in Table 3.1.

`class=` (article/report/book) (default: `article`)
This option selects the base document class. The values load the corresponding KOMA-Script class [5]. The article value thus loads scrartcl.

KOMA-Script is a collection of classes and packages for LaTeX which, in addition to the typographical adaptations to the European area, also greatly extends the configuration options. The documentation is also available in German [6].

`accentcolor=` ⟨*Color*⟩ (default: `0b`)
Selects the spot color for use within the identity bar. In addition to these colors, any defined can be used. See chapter 5 for the list of predefined colors as well as additional options for a more detailed selection. The Options accentcolor, textaccentcolor and identbarcolor will be passed to tudacolors. Using those, the colors can even be set independently of each other.

`custommargins=` (⟨*Boolean*⟩/geometry) (default: `false`)
According to the corporate design, the line lengths are too long from a typographical point of view. This is why the custommargins class option exists. Details are described within subsection 3.6.2.

`marginpar=` (true/false/auto) (default: `auto`)
Controlls the marginpar. The default setting is auto. This means that the marginpar column runs across the fifth column of the layout grid [1]. Above this, the option also accepts boolean values. False would set width of the column to 0. The mechanism itself is not deactivated.

The font inside the marginpars are controlled using the komafont mechanism [5] using the `marginpar` element. It is defined as

```
\setkomafont{marginpar}{\accentfont}
```

to extend this setting, e. g. by adding a color one can add

```
\addtokomafont{marginpar}{\color{textaccentcolor}}
```

to the preamble.

`twocolumn=` ⟨*Boolean*⟩ (default: `false`)
Activates the two-column mode globally. In this case, however, two margin columns are created due to their nature. Use in combination with `marginpar=auto` is therefore questionable in most cases. If the two-column mode is only activated locally, that behavior does not apply, but the margin notes are deactivated.

`ruledheaders=` (all/none/chapter/section) (default: `all`)
Selects the style of the headings. `ruledheaders=all` selects the style framed with rules for all up to

Table 3.1: Differences between the demo files based on tudapub. The template files use different options depending on the actual purpose. This tabular gives an overview of the Features shown inside the tempaltes as well as the initial settings.

Option	DEMO-TUDaThesis	DEMO-TUDaPhD	DEMO-TUDapub	DEMO-TUDaReport
twoside	✘	✔	✘	✘
parskip	✔	✘	✔	✘
colophone	✘	✔	✘	✘
dedication	✘	✔	✘	✘
font size	11pt	11pt	9pt	9pt
ruledheaders	section	chapter	all	all
class	scrreprt	scrbook	scrartcl	scrartcl
thesis	`thesistype=bachelor`	`thesistype=dr, dr=rernat`	✘	✘
marginpar	✘	✘	✔	✘
Affidavit (“Selbstständigkeitserklärung”)	✔	✔	✘	✘
abstract	✘	✔	✔	✘
custommargins	✔	✔	✘	✘

`\subsubsection`. This style is limited accordingly for chapter or section. False loads the KOMA-Script default setup instead.

`title=` (default/small) (default: `default`)
The relatively large font size of the title can lead to spacing issues, especially with long titles for theses. The `title` option prevents this by loading the font sizes set defined for the paper 1 size smaller (e. g. a5 font size if a4 paper is used).

`type=` (publication/thesis/intern) (default: `publication`)
Is used to load specific configuration for theses (section 3.4) or internal documents.

`headline=` (true/false/automark) (default: `false`)
The CI guideline [1] allows headers to be used ruled below the identbar. That may be confusing as the headers are typeset larger than subsection headings. Because of that it’s switched off by default, but users can enable this setting using the `headline` option. `automark` will also be passed to scrlayer-scrpage and switch to running headers.

`logo=` (head/body) (default: `body`)
Sets the position of the logo on the title page, see subsection 3.2.2.

`colorback=` (true/false/title/body/head) (default: `true`)
Change the color setup of the title page, see subsection 3.2.2.

`IMRAD=` ⟨*Boolean*⟩ (default: `true`)
Toggles the check for IMRAD labels, see subsection 3.3.3.

`logofile=` ⟨*file name/path*⟩ (default: `tuda_logo`)
Allows an alternative logo to be used. This option is available so that the templates can also be used without the TUDa logo. The logos are reserved to be used by TUDa members and may therefore not be published with this template. In case you want to install the logo files locally you can find further information at `https://www.ulb.tu-darmstadt.de/finden_nutzen/medien_nutzen/dokumente_erstellen/latex/index.en.jsp`.

Additionally all KOMA-Script options can be used. These are described within the documentation. Some more notes on the interaction between KOMA-Script and TUDa-CI can be found in section 3.7.

3.2 Title

The title page is generated automatically by tudapub. The structure for this corresponds largely to the classic method using `\maketitle`.

The KOMA-Script option `titlepage`[5] is supported to switch between separate title pages and an in-page title block. Due to the implementation, `titlepage=true` is treated identically to `titlepage=firstiscover`.

`\titlehead` `\title` `\subtitle` `\subject` `\author` `\and` `\thanks` `\date` `\publishers`

These macros are used to set the title data. If not mentioned differently all need one mandatory argument which contains the data. They are used as with standard LaTeX, but there were a few added to support the additional features of the title page design.

```
\title{⟨title⟩}
\author{⟨Author1⟩\thanks{Affiliation of Author1} \and ⟨Author2⟩}
% [...other elements...]
\maketitle
```

3.2.1 TUDa-CI specific title elements

`\titlegraphic`

Accepts any LaTeX content, does not have to be an image. This is placed flush with the top corner in the main part of the title page. This macro is usually used to place a graphic:

```
\titlegraphic{\includegraphics[width=\width]{example-image}}
```

`\width` `\height`

`\width` and `\height` can be used to select the appropriate size of the image. From version 3.19, there also is a starred variant `\titlegraphic*`. With this, scaling and any necessary cropping to fill the reserved area is done automatically using trimclip.

```
\titlegraphic*{\includegraphics{example-image}}
```

`\addTitleBox` `\addTitleBoxLogo`

```
\addTitleBox{⟨Box Content⟩}
\addTitleBoxLogo*{⟨Logo⟩}
```

The TUDa CI design is using white boxes to place additional Information or logos on the title page. These can be added using `\addTitleBox` or `\addTitleBoxLogo`.

All title boxes are placed below each other with a specified distance and use a white background. Text or an institute logo may appear here. The background hast to be white.

Whereas `\addTitleBox` will set the box to the same width and horizontal alignment as the TUDa logo the logo variants of this macro don't have a fixed width. Here the default is to place scale the logo automtically and maual scaling has to be enforced using the starred variant:

```
\addTitleBox{Text, e.\,g. Institute}
\addTitleBoxLogo{example-image}
\addTitleBoxLogo*{\includegraphics[width=.3\linewidth]{example-image}}
```

`\AddSponsor`

As of version 3.0, the sponsor mechanism of tudaleaflet is also available in tudapub. This makes it possible to place sponsor logos below the title graphic. Sponsor logos or names can be added using `\AddSponsor`

```
\AddSponsor{⟨Code to insert the Logo or just the name⟩}
```

`\height`

`\height` is defined within the argument. All sponsors added this way are aligned at their baseline. This can be used to scale multiple logos to the same height. The space between the sponsors will be horizontally filled.

`\sponsors`

The second variant enables manual placement with vertical alignment, as may be necessary for logos with different heights. In this case, only the spacing and seperation rules are added around the logos:

```
\sponsors{⟨logo1⟩⟨logo2⟩}
```

For theses there exist additional data fields `\birthplace`, `\group`, `\examdate`, `\submissiondate`, `\tuprints`, `\urn` and `\reviewer`. To use these and also learn about other specials of the title mechanism within theses, please refer to subsection 3.4.1.

3.2.2 Options for titlepage modification

`logo=` (head/body) (default: `body`)
The position of the logo can be switched. This is done via the class option logo;

logo=head The logo is placed in the header directly next to the title, which is reduced in width. The background of the title is colored in the color of the identity bar. This setting also will move the boxes below the logo to the page head.

logo=body The logo including the info boxes is placed in the body of the front page.

`colorback=` (⟨*Boolean*⟩/title/head/body) (default: `true`)
Similar to the logo position the color structure can be adjusted. colorback toggles between the colored block on the title page and the white background.

The other values will enable the colored area but allow switching between positions.

colorback=title Only the title background (without subtitle) is colored.

colorback=head Background of the total title block including subtitles is colored.

colorback=body Only the background of the area used for the `\titlegraphic` is colored.

3.2.3 Extra title material for theses

For mode=thesis the title page works a little different. Here some data elements are used to provide an interface to be overwritten and are prefilled using additional data fields. The fields only existing in thesis mode are: `\birthplace`, `\group`, `\examdate`, `\submissiondate`, `\tuprints`, `\urn` and `\reviewer`. For more information on their use and other settings for theses have a look at section 3.4.

3.2.4 Abstract

tudapub extends the availability of the

`abstract` (*env.*) environment to scrbook. Additionally, it adds an optional argument to select the language. This should be used to use multiple abstracts of different languages within one document.

```
\begin{abstract}
  Abstract using the document main language (here English)
\end{abstract}

\begin{abstract}[german]
  Weitere Zusammenfassung in einer anderen Sprache (hier Deutsch),
  sofern benötigt.
\end{abstract}
```

It is important that all languages used in the document are loaded. In the case of the example, both options, english and german must be passed to the babel package.

3.3 PDF/A

`pdfa=` The university library requires submissions to be compliant to the PDF/A standard. (⟨*Boolean*⟩)
The TUDaPub class supports the PDF/A 2b standard. PDF/A mode is automatically activated within tudapub. Depending on the document it's using an implementation via the pdfx package or the LaTeX kernels own PDF management.

LaTeX is not validating the file in any way, it is just using compatible settings for all elements processed by the compiler. As LaTeX is not doing any processing on image files the user has to ensure the font settings match the requirements for PDF/A.

`pdfx=` (⟨*Boolean*⟩) (default: `true`)

tudapub will try to automatically select the best method. This is done using pdfx option. This should only be toggled manually if the user is totally sure to understand the impact. Please be aware that the pdfx option is incompatible with any use of the pdfmanagement. There will be an error message if this combination is active and pdfx as well as pdfa will be disabled.

3.3.1 Creating PDF/A using the PDF management

The template files which use PDF/A by default now include the settings for the PDF management:

```
\DocumentMetadata{
  pdfstandard=a-2b,
  pdfversion=1.7,
  lang=en,% or de or …
}
```

This can also be used within other documents. More Information on this structure can be found in the documentation of the pdfmangement-testphase [18] as well as l3pdfmeta [17].

If this setting is used tudapub will automatically disable the loading for pdfx. tudapub will automatically try to pass the title data to the metadata. In case the title contains more complex material which can not be expanded into text it's possible to overwrite these settings using hyperref's `\hypersetup`. The demo files include examples.

3.3.2 Creating PDF/A using the pdfx

In older versions of this bundle we used pdfx to create PDF/A compliant output files. If the PDF management using `\DocumentMetadata` is not detected to be active, tudapub will still try to use this mechanism. The metadata setup for the XMP data then works completely different.

The compiler is creating an additional `\jobname.xmpdata` file. It is tried to insert the title data directly. The title data is usually transferred directly. However, this can lead to problems if there are macros used within title material. For example, the subtitle for this document contains the LaTeX macro, but only text elements can be used. Similar to the link labels within the PDF bookmarks.

`\Metadata` To avoid this issue, tudapub provides the macro `\Metadata`. All variables that can be processed by pdfx can be set here according to the key=value structure.

```
\Metadata{
  author=Marei Peischl (peiTeX),
  title=LaTeX im Corporate Design der TU Darmstadt,
}
```

Please note that this macro only works if PDF/A output is activated, and no PDF management is used. If this is not the case, tudapub issues a corresponding error message or warning.

For a full list of the available metadata fields have a look at the pdfx documentation [15].

3.3.3 Additional metadata as requested by the university library

There is a mechanism for identifying the IMRaD [3] structuring model. In the style of the individual sub-areas, the call of

```
\IMRADlabel{introduction}
\IMRADlabel{methods}
\IMRADlabel{results}
\IMRADlabel{discussion}
```

would generate the corresponding labels. They have the name `IMRAD:⟨key⟩`.

`IMRAD=` ⟨*boolean*⟩ (default: `true`)
The check mechanism is activated by default at the request of the library. If you are not planning to use these labels or this structure just don't match your document, the warning can be deactivated using the IMRAD=false.

3.3.4 Colors conversion and profile

PDF/A can only use one color model within a document. By default, if no specific profile is selected TUDa-CI will convert the colors to RGB. However, as there is no clear conversion, CMYK elements should not be used in pdfa=true mode.

This mode is not suitable for print output. There will be a warning if the conversion is triggered. To use CMYK colors with PDF/A it's necessary to use PDF/A via the PDF management (subsection 3.3.1) and select a specific color profile [see 17].

3.4 (PhD) Theses – Special options and elements used in these templates

As mentioned before the these templates `DEMO-TUDaThesis` and `DEMO-TUDaPhD` are based on tudapub. Therefore, they support all options as described before but provide some additional mechanisms. To load the thesis specific config the option type=thesis has to be used. As there are additional thesis specific options it's also possible to use the thesis key directly and add the specific options within braces:

```
\documentclass[
english,
class=report,
thesis={type=master}
[…]
]{tudapub}
```

`thesis/type=` bachelor/pp/master/dr/drfinal/⟨*text*⟩ (default: `initially unset`)
Selection of the type. This is printed on the title page and also selects which data is mandatory. Possible values are listed below including their impact. The items in parentheses list the required data for that type.

sta "Studienarbeit": Student research project. (`\title`, `\author`, `\date`)

bachelor "Bachelor thesis". (`\title`, `\author`, `\submissiondate`, `\department`, `\reviewer`)

master "Master thesis". (`\title`, `\author`, `\submissiondate`, `\department`, `\reviewer`)

pp "Project proposal". (`\title`, `\author`, `\date`, `\department`)

dr submitted Doctoral thesis (`\title`, `\author`, `\submissiondate`, `\department`, `\reviewer`)

drfinal accepted Doctoral thesis (`\title`, `\author`, `\submissiondate`, `\examdate`, `\department`, `\reviewer`)

If a type is specified that was not listed the text is directly used as type. In this case, there are no mandatory title fields except the title.

`dr=` (rernat/rerpol/ing/phil) (default: `initially unset`)
Loads one of the predefined texts for the title page.

For example, the value phil would use: „Zur Erlangung des Grades eines Doktor der Philosophie (Dr. phil.)" Please be aware that the titlepage is enforced to use German.

`\drtext` If none of these values correspond to the desired title, a text can be transferred directly.

```
\drtext{To obtain the degree …}
```

`department=` (⟨*shorthand or text*⟩) (default: `initially unset`)
The departments are permanently stored as text modules in German and English. This option enables selection as a document class option. For compatibility reasons, however, the department macro can also be used for this. The following shorthands are predefined:

arch → Architecture
bauing → Civil and environmental engineering
bio → Biology
chem → Chemistry
etit → Electrical Engineering and Information Technology
gugw → History and Social Sciences
humanw → Human Sciences
inf → Computer Science
mb → Mechanical Engineering
matgeo → Materials and Earth Sciences
math → Mathematics
phys → Physics
wi → Law and Economics

In addition to the departments, there are also "fields or study". These are not available for doctoral theses. If the given value is not found in the departments the fields of study will be checked afterwards. The study areas have the following shorthands:

ce → Computational Engineering
ese → Energy Science and Engineering
ist → Information Systems Engineering
mech → Mechanics
metro → Mechatronics

If anything other than one of these shorthands is found, the provided text is used directly and a corresponding warning is issued.

3.4.1 Thesis specific title page settings and customization

`instbox=` ⟨*Boolean*⟩ (default: `true`)
Selecting the department also creates a box on the title page below the logo. In some cases this setting does not match the requirements. Therefore, the option instbox can be deactivated.

`ignore-missing-data=` (⟨*Boolean*⟩) (default: `false`)
This option is a switch that makes it possible to disable the error message about title data that has not been transferred. In this case, only a warning is created if the specified data does not match the requirements.

`\publishers` Is used here for the location and is preset with „Darmstadt“ or „Darmstadt, Technische Universität Darmstadt“ (for PhD theses).

`\subject` Will be placed below the `\subtitle` as for the normal title page, but will hold all the additional information which can be provided using the data fields described in this section.

`\birthplace` `\department` Place of birth. Subject/field of study. It's preferred to set it via the class option. However, the argument is processed in the same way. This macro also provides the functionality of specifying entries that differ from the default entries. Especially if a different text to the default "in the department of" and its variants is required. For this purpose, `\department` provides an optional argument:

```
\department[⟨replacement text⟩]{⟨abbreviation/name⟩}
```

In addition, from version 2.01 there is also the option of replacing the entire text "in the department ⟨*department name*⟩" as well as the information in the info box on the title page. This is done using the starred variant:

```
\department*[text for the box]{text between type and author}
```

`\examdate` Will be added within the description block `\subject`.

`\submissiondate` `\institution` `\department` `\institute` `\group` `\reviewer` Will be added inside a title box as was described in section 3.2 below each other.

Reviewers. Multiple reviewers are separated by `\and` as for authors. The numbering runs from left to right.

Adjusting the reviewer labels

The identifier can be changed using an optional argument:

```
\reviewer[⟨replacement identifier⟩]{⟨name1⟩ \and ⟨name2⟩}
```

To change the numerical designation, a comma list is used instead of a single label:

```
\reviewer*[identifier1, identifier2]{name1 \and name2}
```

In this case, the automatic numbering before the identifier is omitted. If, for example, the wording of the doctoral regulations is to be complied with, the following applies:

```
\reviewer[Erstreferent\_in,Koreferent\_in]{Name1 \and Name2}
```

`\setupReviewName` There is also a macro for creating department-specific templates. This allows changes to be made without calling `\reviewer`.

```
\setupReviewName{⟨Alternative label to replace "Review"⟩}
\setupReviewName[1]{⟨Erstreferent⟩}
\setupReviewName*{⟨label1⟩,⟨label2⟩}
```

`reviewer-on-uppertitleback=` ⟨*Boolean*⟩ (default: `false`)
As of version 3.26, the reviewers are no longer named on the back of the title page. This can be controlled using the thesis option reviewer-on-uppertitleback.

`\studentID` Matriculation number. According to the template specifications, this information is always optional.

`\titleintro` `\titleaddendum` From version 2.03, these hooks can be used to add any text directly in front or after the automatic generated subject block holding the thesis data.

`\tuprints` Publication via TUprints requires this setting. The data should be provided by the TUprints submission process.

```
\tuprints{
  doi=https://doi.org/10.26083/tuda-1234,
  urn=https://nbn-resolving.de/urn:nbn:de:tuda-tuda-567890,
  url=https://tuprints.ulb.tu-darmstadt.de/handle/tuda/56789,
  year=2022
}
```

`printid=` ⟨*TUprints print ID*⟩ (default: ⟨*initially unset*⟩)
`url=` ⟨*TUprints print ID*⟩ (default: ⟨*initially referencing the printid*⟩)
`urn=` ⟨*TUprints URN*⟩ (default: ⟨*initially unset*⟩)
`doi=` ⟨*TUprints DOI*⟩ (default: ⟨*initially unset*⟩)
If the argument does not contain an equals sign, the value is set as printid and no urn is specified.

printid is the ID number of the TUprints entry and will be used automatically to create a hyperlink to the corresponding URL. The URN is – in addition to the DOI – a permanently unique resource identifier for the document. In TUprints, the number corresponds to the printid with the addition of a check digit. All dates can be found in the details of the TUprints entry.

License information

`license=` ⟨*License key or license text*⟩ (default: `cc-by-4.0`)
From version 3.08 there are predefined values for license simplify customization. These are as follows:

`cc-by-4.0` default since version 4.0	`cc-by-nd-4.0`
`cc-by-sa-4.0`	`cc-by-nc-nd-4.0`
`cc-by-nc-sa-4.0`	`inc-1.0` (From version 3.36)
`cc-by-nc-4.0`	`cc-by-nc-nd-2.0-en`

The introduction of this option was part of the preparation for adapting the standard license. The corresponding discussion can be found at https://github.com/tudace/tuda_latex_templates/issues/251. The adjustment of the default setting for TUDa-CI was made with version 4.0.

The ULB of TUDa offers support in choosing a suitable Creative Commons license at https://www.ulb.tu-darmstadt.de/dpub or the CC project itself via its license finder at http://creativecommons.org/choose/. TU Darmstadt recommends use of the open CC BY 4.0 license in its Publication Guidelines and Open Access Policy.

If a value different from the keys listed above is found, this value is used directly in place of the license text. If it itself contains an equals signs or commas, grouping is necessary.

3.4.2 Affidavit

`\affidavit` The macro `\affidavit` creates a declaration of authorship with a signature line. The name/signature could be set via options or the Information provided with `\author` is used. In the demo documents, the affidavit is located directly after the title.

`hide-architecture-note=` ⟨*Boolean*⟩ (default: `false`)
The class option hide-architecture-note allows to disable the note specifically tagetting theses of the architecture department. For compatibility reasons it's false by default. So the note will be printed. It's also possible to set this directly as an affidavit option if it's not a PhD theses.

Disclaimer:

The `\affidavit` command is using the current version (as of 2025-01-22) of the text provided at https://www.tu-darmstadt.de/studieren/studierende_tu/studienorganisation_und_tucan/hilfe_und_faq/artikel_details_de_en_37824.de.jsp.

According to the department II, the legally binding text can always be found there. The docx file should be used, printed out, signed, scanned and then integrated. The easiest way to do this is to use the pdfpages package. Please ensure to check which regulations apply to your thesis before submission. TUDa-CI can not ensure the text is up to date or matches your submission type.

From version 3.32, the distinction between an affidavit for digital or printed submissions, which has been supported since version 3.06, no longer applies. For compatibility reasons, the options are still available, but have the same effect. It is imperative that students check whether the text corresponds to the required version before submitting a thesis.

PhD theses use a different text here; the affidavit option affidavit=dr is used internally to differentiate between them.

Version 3.20 also allows the transfer of further options for the signature name, a signature image or the location. The extent to which these options may be used must be clarified by the user before submission. TUDa-CI cannot make a reliable statement on this.

```
\affidavit[
  signature=Signature name,
  signature-image={\includegraphics[width=\width]{signatureimage}}
]
```

A vertical shift of the signature image is not implemented directly, but is easily possible by using the LaTeXmacro `\raisebox{`⟨*shift*⟩`}{`⟨*content*⟩`}`.

It is also possible to print an affidavit in another language as a supplement. In order to handle the structure and any necessary language switching, there is an environment:

```
\begin{affidavit*}[⟨Babel language option⟩]{⟨heading⟩}
  \meta{Text}
\end{affidavit*}
```

This version deliberately does not have a signature line, as the developers do not consider this version to be legally binding. However, the environment can also be used for special forms of explanation. In this case, an additional signature line can be added: ⟨*Location*⟩

`\AffidavitSignature`
`signature-location=`

```
\AffidavitSignature[⟨city⟩]
```

3.5 Further configuration options

3.5.1 Select the color model

Color vision depends on the way colors are presented. For professional printing there usually is a color profile required by the printer to be used. For use in documents, it is therefore important to know which output medium is to be used primarily. Technically, this difference is reflected in color mixing models. In accordance with the guidelines, TUDa-CI supports both a color model for print output (cmyk) and for screen display (RGB). The implementation is done via the xcolor package, whereby the corresponding color values for both models are stored withing tudacolors.

Normally, TUDa-CI automatically selects a suitable model. The default setting of pdfa=true ensures a conversion to RGB (vgl. 3.3.4) if no manual setting was found.

If a specific color profile should be used, the xcolor options cmyk or RGB can be transmitted directly to tudapub. They are passed on to the package and will be processed according to the xcolor documentation [4].

3.6 Customizations that deviate from the corporate design guidelines

3.6.1 Font size

`fontsize=` (⟨*length*⟩) (default: `9pt`)

Contrary to the corporate design guidelines, tudapub can also process other font sizes. The fontsize option of KOMA-Script is supported (e. g. fontsize=11pt). If no special font size configuration file is available for TUDa-CI, the file supplied with KOMA- Script is selected. Examples of deviations for typographical reasons are also shown in the demo files for theses.

3.6.2 Margins

`custommargins=` (true/false/geometry) (default: `false`)

According to the corporate design, the line lengths are too long from a typographical point of view. This is why the custommargins class option exists.

custommargins=false Default setting of tudapub. The margins correspond to the specifications of the Corporate Design Guidelines [1]. The setting is made using geometry. Customizations are overwritten by executing `\maketitle`. To allow personal adjustments one of the other settings is required.

custommargins=true The settings of the Corporate Design Guidelines are not activated. pkggeometry is not loaded. This mode corresponds to the default setting of KOMA-Script. The margins are not explicitly defined, but calculated on the basis of the typearea package [see 5].

custommargins=geometry This variant was created based on user requests. It allows using tudapub with support for manual adjustments. geometry is loaded and preconfigured as with custommargins=false. However, it is possible to make minor adjustments by using the `\geometry` command. The settings that apply at the start of the document are saved and restored after the title pages.

Please note that the settings use the preset type area as a starting point (with or without a margin column, depending on the option). It is possible to reset all options before adding your own:

```
\geometry{
  reset,
  ⟨Adjustments starting from geometry's defaults⟩
}
```

This applies in particular to the options `includehead`, `includefoot` and `includemp`.

Remark on the headers/footers

If the option marginpar=true remains set, the header and footer protrude beyond the margin column. For aesthetic reasons, it is therefore recommended in this case to limit the header and footer to the text area with marginpar=false.

The standard layout of the column titles is also not very advantageous, as the column titles can be locally larger than the actual headline. (headline=automark)

For this reason, tudapub provides a simpler page style, which considerably simplifies use with living column titles. The structure is realized using scrlayer-scrpage and can be adapted according to the KOMA-Script documentation [5].

```
\pagestyle{TUDa.headings}
```

Remark on binding correction

`BCOR=` (⟨*length*⟩) (default: `0pt`)
`BCORtitlepage=` (⟨*Boolean*⟩) (default: `false`)
If a binding correction (BCOR=⟨*length*⟩) is used, this is not automatically inserted on the title page. For this case, the BCORtitlepage option was added with version 3.0. If this is activated, the title page uses the value of the typearea option BCOR on the first page as an addition to the left margin.

3.6.3 Frontmatter/Mainmatter/Backmatter

The macros `\frontmatter`, `\mainmatter` and `\backmatter` are usually only available for the class scrbook. On request, these macros have also been provided as a basis for scrartcl and scrreprt. It is therefore possible to switch to Roman numerals for the opening credits. Arabic numerals are then used from `\mainmatter`.

3.6.4 Math fonts

As there is no compiler-independent universal math font and the corporate design guidelines do not take any recommendations into account, several possible variants were discussed. The default setting always corresponds to the installation standard. No specific settings are loaded. The discussion on this can be found at: https://github.com/tudace/tuda_latex_templates/issues/19

A few example configurations are shown below. In principle, however, the math type is freely selectable – apart from the restrictions of the compiler. In many cases, the "TEX Font Catalogue" is helpful for selection and use: https://tug.org/FontCatalogue/mathfonts.html

pdfLaTeX

For pdfLaTeX there is a solution which combines the letters of the default Text font (Charter) with math symbols of different fonts.

```
\usepackage[charter,expert]{mathdesign}
```

There are similar approaches for a few other combinations. Some examples can be found in the XCharter documentation [16]. http://mirrors.ctan.org/fonts/xcharter/doc/xcharter-doc.pdf

3.7 Customization using KOMA-Script

As the class is fully KOMA-Script-compatible apart from a few forced settings that affect the layout, a look at the KOMA-Script documentation [5] is helpful for any kind of modifications. For most of the possible modifications KOMA-Script offers its own solutions, which often makes supplementary packages superfluous. Examples of typical modifications that are also permitted as part of the corporate design:

- Change paragraphs to use a skip instead of an indent of the first line (option parskip)
- Element numbering with or without end dot (option numbers=enddot/noenddot)
- Caption positioning, alignment and spacing (The macros `\captionabove`, `\captionbelow`, `\captionof` and the captions option)

3.8 Known issues and incompatibilities

3.8.1 XeLaTeX and PDF/A

If the pdfx package is used together with XeLaTeX for the creation of PDF/A the support is limited. There will be a corresponding warning. With some XeLaTeX versions it's possible that there even may be errors. LuaLaTeX should be preferred, but in the worst case switching to the pdfmanagement mechanism could also help.

3.8.2 DVI Output

Due to the default setting for creating a PDF/A file, it is not possible to use TUDa-CI in standard settings to create a DVI file. However, a large part of the functionality can be used when pdfa mode is deactivated.

3.8.3 Possible option clash for microtype

The microtype package is loaded automatically if pdfLaTeX is used, as the ligatures for small caps must be deactivated in the font to avoid problems (see https://github.com/tudace/tuda_latex_templates/issues/144). It is possible to pass further options to microtype before loading the document class:

```
\PassOptionsToPackage{⟨microtype options⟩}{microtype}
\documentclass{tudapub}
```

4 Department-specific adaptations

Some departments have special requirements. TUDa-CI includes an interface to be extended in that way. Currently the only official extension is the one for the department of mechanical engineering. However, the mechanism can be expanded.

4.1 Department of Mechanical Engineering

The corresponding mode is activated via the option department=mecheng. Coloring is adjusted automatically and additional layout elements like the „Zeitstrahl“ are introduced. The modifications exist for all document types provided by TUDa-CI.

`\SetPaperID` In addition, some documents require the placement of document identifier on the titlepage. For this purpose `\SetPaperID` was introduced.

`\SetPaperID{`⟨*Letter*⟩`}{`⟨*ID*⟩`}`

This also works without activating mecheng mode. However, the option adds some additional parameters for customized distances.

The mode also sets the options: colorback=false and ruledheaders=section.

Department logo

`departmentlogofile=` ⟨*filename*⟩ (default: `tuda_maschinenbau_logo`)
The department logo can be downloaded and installed the same way as the TUDa logo. A different file can also be selected using the option. If the value remains empty, no image is inserted.

Colors

The department subdivides the colors described in the corporate design manual. Therefore, if department=mecheng has been activated, the following color names also exist:

```
\colorlet{TUDa-Primary1}{TUDa-6b}
\colorlet{TUDa-Primary2}{TUDa-2d}
\colorlet{TUDa-Secondary1}{TUDa-9a}
\colorlet{TUDa-Secondary2}{TUDa-8a}
\colorlet{TUDa-Secondary3}{TUDa-6a}
\colorlet{TUDa-Secondary4}{TUDa-3a}
\colorlet{TUDa-Secondary5}{TUDa-4a}
\colorlet{TUDa-Secondary6}{TUDa-5a}
\colorlet{TUDa-Arrow}{TUDa-Primary2}
```

„Zeitstrahl“ arrow

`\MechEngArrow` The design element of the timeline can be created using the macro `\MechEngArrow{`⟨*length*⟩`}`. The color corresponds to the color `TUDaArrow`, which is pre-assigned with the second primary color (blue).

5 tudacolors –– Color definitions

The tudacolors package provides the general color definitions necessary for the TUDa-CI Templates. It defines the colors according to the design guideline [1]. The definitions are using the xcolor package [4]. Additionally the mecheng specific colors (see subsubsection 4.1) but the additional colors used by the new tudabeamer layout are provided directly in the color theme. This is because right now these colors are only defined in RGB mode and should not be used in documents using CMYK colors.

The colors can be used like within standard LaTeX. The predefined TUDa-CI specific colors are:

TUDa-0d	TUDa-9a	TUDa-10b	TUDa-11c
TUDa-0c	TUDa-10a	TUDa-11b	TUDa-1d
TUDa-0b	TUDa-11a	TUDa-1c	TUDa-2d
TUDa-0a	TUDa-1b	TUDa-2c	TUDa-3d
TUDa-1a	TUDa-2b	TUDa-3c	TUDa-4d
TUDa-2a	TUDa-3b	TUDa-4c	TUDa-5d
TUDa-3a	TUDa-4b	TUDa-5c	TUDa-6d
TUDa-4a	TUDa-5b	TUDa-6c	TUDa-7d
TUDa-5a	TUDa-6b	TUDa-7c	TUDa-8d
TUDa-6a	TUDa-7b	TUDa-8c	TUDa-9d
TUDa-7a	TUDa-8b	TUDa-9c	TUDa-10d
TUDa-8a	TUDa-9b	TUDa-10c	TUDa-11d

tudacolors provides package options to allow color selection or the color mode. Since tudacolors is usually not loaded directly but indirectly using one of the TUDa-CI document classes, the options can be added as class options, as described in section 3.1. They will be passed to the package.

`accentcolor=` ⟨*Color*⟩ (default: `0b`)
`accent=`

Highlight color used for all highlight elements by default. The options below will reference this color as default setting metaaccentcolor. Depending on the documentclass it might be used for the so-called "identbar" as well as highlighted test elements, e. g. within letters if premium is enabled.

|accent| is an alias. We recommend using |accentcolor| to simplify recognizing the purpose of that option.

`textaccent=` ⟨*color*⟩ (default: ⟨`accentcolor`⟩)
The accent color for highlighted text. Usage depends on the document class.

`identbar=` ⟨*color*⟩ (default: ⟨`accentcolor`⟩)
Color of the colored bar on the top of the page or slides using the 2008 layout.

`text=` preferblack/preferwhite (default: `preferwhite`)
The text placed on top a colored area, e. g. if the background of the title is set colored is either set to black or white. There are some color combinations forbidden according to the design guideline [1] and some colors are allowed to use black or white. This option can be used to prefer either black or white for the title in case both variants are allowed. It's not possible to enforce a forbidden setting using this option.

`colormode=` cmyk/RGB Depends on document type
The colormode can be selected to overwrite the default setting. Please be aware, that there is

a difference in casing and `RGB` has to be used uppercase and `cmyk` lowercase. The reason for this is that internally uppercase and lowercase model names slightly differ because of rounding.

`department=` default/mecheng/⟨*department*⟩ (default: `default`)
Within tudacolors it's only checked if the department was set to mecheng. In that case the additional color configuration will be enbled (section 4.1).

Change History

v3.90
General: Converted to DTX file 1

Bibliography

[1] *Das Bild der TU Darmstadt. Corporate Design Handbuch*. June 2011. URL: https://www.intern.tu-darmstadt.de/media/medien_stabsstelle_km/services/medien_cd/das_bild_der_tu_darmstadt.pdf (visited on 05/01/2020).

[2] *GitHub Repository: TUDa-CI – Corporate Design LaTeX-Templates for TU Darmstadt*. URL: https://github.com/tudace/tuda_latex_templates/ (visited on 01/22/2025).

[3] *IMRAD*. Wikipedia. URL: https://en.wikipedia.org/wiki/IMRAD (visited on 12/20/2018).

[4] Uwe Kern and LaTeX Project. *The `xcolor` package. Driver-independent color extensions for LaTeX and pdfLaTeX*. Version 3.02. Sept. 29, 2024. URL: http://mirrors.ctan.org/macros/latex/contrib/xcolor/xcolor.pdf.

[5] Markus Kohm. *KOMA-Script. Die Anleitung*. Jan. 14, 2019. URL: http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide-en.pdf (visited on 08/03/2024).

[6] Markus Kohm. *KOMA-Script – Ein wandelbares LaTeX-Paket. Die Anleitung*. June 16, 2023. URL: http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide-de.pdf (visited on 08/03/2024).

[7] *LaTeX – ULB TU Darmstadt*. URL: https://www.ulb.tu-darmstadt.de/finden_nutzen/medien_nutzen/dokumente_erstellen/latex/index.en.jsp (visited on 01/22/2025).

[8] Marei Peischl. *LaTeX beamer using TU Darmstadts Corporate Design*. Jan. 26, 2025. URL: http://mirrors.ctan.org/macros/latex/contrib/tuda-ci/doc/DEMO-TUDaBeamer.pdf.

[9] Marei Peischl. *tcolorbox-Poster using TU Darmstadt's Corporate Design*. Jan. 26, 2025. URL: http://mirrors.ctan.org/macros/latex/contrib/tuda-ci/doc/DEMO-TUDaSciPoster.pdf.

[10] Marei Peischl. *The TUDa-CI package. LaTeX templates for Technische Universität Darmstadt*. Jan. 24, 2025. URL: https://ctan.org/pkg/tuda-ci.

[11] Marei Peischl. *TUDaLeaflet*. Jan. 26, 2025. URL: http://mirrors.ctan.org/macros/latex/contrib/tuda-ci/doc/DEMO-TUDaLeaflet.pdf.

[12] Marei Peischl. *TUDaLetter – LaTeXletters using TU Darmstadt CI*. Jan. 26, 2025. URL: http://mirrors.ctan.org/macros/latex/contrib/tuda-ci/doc/DEMO-TUDaThesis.pdf.

[13] Marei Peischl. *Übung zur Benutzung der TUDaExercise-Klasse*. Jan. 26, 2025. URL: http://mirrors.ctan.org/macros/latex/contrib/tuda-ci/doc/DEMO-TUDaExercise-de.pdf.

[14] Marei Peischl. *LaTeX using TU Darmstadt's Corporate Design*. Jan. 26, 2025. URL: http://mirrors.ctan.org/macros/latex/contrib/tuda-ci/doc/DEMO-TUDaPoster.pdf.

[15] C. V. Radhakrishnan et al. *Generation of PDF/X- and PDF/A- compliant PDFs with pdfTeX – pdfx.sty*. July 1, 2024. URL: https://mirrors.ctan.org/macros/latex/contrib/pdfx/pdfx.pdf (visited on 07/22/2024).

[16] Michael Sharpe. *The XCharter Font Package*. June 18, 2024. URL: http://mirrors.ctan.org/fonts/xcharter/doc/xcharter-doc.pdf (visited on 08/04/2024).

[17] The LaTeX Project. *The l3pdfmeta module. PDF standards*. May 23, 2024. URL: http://mirrors.ctan.org/macros/latex/contrib/pdfmanagement-testphase/l3pdfmeta.pdf (visited on 07/22/2024).

[18] The LaTeX Project. *The LaTeX PDF management testphase bundle. PDF standards*. May 23, 2024. URL: http://mirrors.ctan.org/macros/latex/contrib/pdfmanagement-testphase/pdfmanagement-testphase.pdf (visited on 07/22/2024).

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.