The mucproc class*

Marei Peischl
tex@mareipeischl.de

February 14, 2017

1 Introduction

The mucproc class provides a document class which allows to use the LaTeX document preparation system for contributions to the German "Mensch und Compuer 2017" conference. It corresponds to the Word-template offered via the conference's homepage `http://muc2017.mensch-und-computer.de/` for the year 2017. By offering this document class and an additional usage demonstration file (`mucproc_demo.tex`) we want to offer an easy to use possibility to prepare conference contributions in LaTeX.

The goal of this document class is to create PDF-files which look like those created with the Word-template. Most of the contributions are still expected to be typeset in Word, that's why the mucproc class uses the historically grown formatting guidelines which have been created for Word-Users. Therefore the settings of this document class might not always be the typographically best.

2 Usage

A complete example of use can be found on the github repository for this project: `https://github.com/Blubu/mucproc/`

mucproc already configures all features needed for the usually required elements of a contribution to the MuC-conference. It loads a bunch of packages to simplify this process. Though the Syntax of the document using this class doesn't differ from classic LaTeX documents.

2.1 Class options

`\documentclass` Start your document by loading the mucproc class with an option for the document's main language. The class will expect either `ngerman` or `english` as main language. Otherwise an error message will be thrown.

*This document corresponds to mucproc v1.01 MuC-conference, dated 2017/02/22.

For the language settings this class uses the babel package and loads csquotes for context sensitive quotations. The language option will also be used to configure the bibliography.

The other class option which can be used by authors is `anonymous=`*`truth value`*. It activates the anonymous-mode which shall be used for the submission to the review process before a contribution may be accepted. This option disables the author-field for the title and the additional author information as will be described later.

The third type of class option is not meant to be used by the authors themselves. It rather provides a possibility to force a document to be compiled in the final mode and enables some additional features of the document class. The option `forcefinal=`*`truth value`* will be used for a recompilation process for the preparation of the final version for the publication process of the proceedings. It can be passed to the class via command line interface and will disable a manually set anonymous-mode.

2.2 Title creation

`\title` The `\title` command behaves identically to most popular document classes. There may be line breaks created with `\\` or `\newline` inside the argument. Other formatting macros shall not be used, since this argument is also used for the metadata of the final PDF-file.

`\author` `\and` `\thanks` The `author` macro can also be used similar to standard LaTeX documents. If the document has more than one author, the names shall be separated with `\and`. The `\thanks` command shall be used to add the corresponding institution. It can be used similar to the standard LaTeX definition, though it also can process an optional argument. The optional argument takes a label to reference an institution of another author. This prevents multiple mention of institutions. This macro is used to reference the labels set with the `\thanks` command. The implementation of the institution reference is very different from the usual label/ref mechanism. That's why the reference is only defined after the label was set. If the argument of `\thanksref` isn't a known Label an error will occur.

`\thanksref`

```
\author{John Doe\thanks[inst:1]{Institution 1}\and
   Jane Doe\thanks{Institution 2}\and
   John Doe II\thanksref{inst:1}
```

In contrast to the standard classes, mucproc doesn't place the footnotes created by `\thanks` on the bottom of the page, they are positioned directly below the author field of the title.

Other possible entries for the title (e. g. `\date`)are ignored.

`\maketitle` The title has to be created by calling the `\maketitle` command as usual.

2.3 Additional author information

For the final version the authors of contributions have the possibility to add additional information about all authors. For this the mucproc class provides the `authoraddendum` environment.

`authoraddendum`

This environment can take up to 2 arguments, one optional containing the path to a corresponding photograph and a mandatory one for the Authors name.

```
\begin{authoraddendum}[john_doe.png]{Doe, John}
Text describing the author.
\end{authoraddendum}
```

If the optional picture argument ist left empty the text will be typeset over the whole `\textwidth`, otherwise there will be a 3 cm column on the left side containing the photograph.

3 Implementation

This section contains the source code to the mucproc.cls. It will be tried to explain how the class breaks the tasks.

3.0.1 Compatibility mode

This document class intends to be compiled on as many different distributions as possible. Basically this class compiles on releases newer than 2014/05/01 without errors. Nevertheless some specific fallback tests require a the expl3 package version of 2015/09/07 or newer. A compatibility mode test is implemented first and deactivates all problematic features of the class.

```
\RequirePackage{expl3}
\ExplSyntaxOn
\bool_new:N \g_MUC@compatibility_mode
\@ifpackagelater{expl3}{2015/09/07}{
\bool_gset_false:N \g_MUC@compatibility_mode
}{
\msg_new:nnn {mucproc}{compatibility}{
    Your~ LaTeX-Distribution~ seems~ to~ be~ outdated.\\\\
    Enabling~ compatibility~ Mode.\\\\
    Some~ features~ won't~ be~ available.
    }
    \msg_warning:nn {mucproc}{compatibility}
\bool_gset_true:N \g_MUC@compatibility_mode
}

\RequirePackage{xparse}
```

The boolean \g_MUC@compatibility_mode is set true, if the expl3 is older than required.

3.1 Class options

The key-value class options are implemented using the scrbase package defining a Family called MUC.

```
\RequirePackage{scrbase}
\DefineFamily{MUC}
\DefineFamilyMember{MUC}
\bool_new:N \g_MUC@pdfa_bool
```

The Options anonymous and forcefinal are implemented using the basic scrbase mechanism for boolean options:

```
\FamilyBoolKey{MUC}{anonymous}{MUC@anonymous}
\FamilyBoolKey{MUC}{forcefinal}{MUC@final}
\FamilyBoolKey{MUC}{headline}{MUC@headline}
\FamilyBoolKey{MUC}{copyrightinfo}{MUC@copyrightinfo}
\FamilyBoolKey{MUC}{pagestyle}{MUC@pagestyle}
```

Set up the default boolean values before processing options.

```
\MUC@headlinetrue
\MUC@copyrightinfotrue
\MUC@pagestyletrue

\FamilyProcessOptions{MUC}
```

The pagestyle settings can be disabled if headline and copyrightinfo are set false.

```
\ifMUC@headline
\else
    \ifMUC@copyrightinfo
    \else
        \MUC@pagestylefalse
    \fi
\fi
```

If the forcefinal option is set, all global options containing the substring “draft” or both of the substrings“final” and “false” are removed to avoid any global draft mode.

```
\ifMUC@final
    \bool_if:NTF \g_MUC@compatibility_mode {
        \msg_new:nnn {mucproc} {compatibility-final-error} {
        forcefinal~ is~ not~ possible~ in~ compatibility~ mode.\\\\
        Update~ your~ TeX~ distribution~ and retry!}
        \msg_error:nn {mucproc} {compatibility-final-error}

    }{
    \sys_if_engine_luatex:F {
        \msg_new:nnn {mucproc} {luatex-required} {
```

```
        forcefinal~ requires~ LuaLaTeX~ for~ correct~ PDF/A~ output.\\\\
        Remove~ forcefinal~ option~ or~ change~ the~ compiler.
        }
        \msg_error:nn {mucproc} {luatex-required}
    }
    \clist_new:N \l_MUC@globalopts_clist

    \msg_new:nnn {mucproc} {force-final} {
        forcefinal~ option~ is~ enabled.\\\\
        Ignoring~ all~ global~ complementary~ options.}
    \msg_warning:nn {mucproc} {force-final}
    \msg_new:nnn {mucproc} {remove-draft}{
        forcefinal~ enabled:~ The~ global~ Option~ #1~ is~ ignored,~
        because~ it~ seems~ to~ activate draft~ mode!}

    \exp_args:Nx \clist_map_inline:nn {\@classoptionslist} {
     \tl_if_in:nnTF {#1} {draft}
            {\msg_warning:nnx {mucproc} {remove-draft} {#1}}
            {\clist_put_right:Nn  \l_MUC@globalopts_clist {#1}}
        \tl_if_in:nnTF {#1} {final}
            {\tl_if_in:nnTF {#1} {false}
                {\msg_warning:nnx {mucproc} {remove-draft} {#1}}
                {\clist_put_right:Nn  \l_MUC@globalopts_clist {#1}}
            }
            {\clist_put_right:Nn  \l_MUC@globalopts_clist {#1}}
    }


    \edef\@classoptionslist{\clist_use:Nn \l_MUC@globalopts_clist {,}}
```

Additionally enable PDF/A-mode and metadata generation and disable anonymization.

```
    \bool_gset_true:N \g_MUC@pdfa_bool
    \MUC@anonymousfalse
    }
\fi
```

3.2 Basic Setup

The next step is loading scrartcl as base class. According to the formatting rules we pass some options and use the `mucfontsize10pt.clo` file for setting up the font sizes.

```
\newcommand*{\@fontsizefilebase}{mucfontsize}
\PassOptionsToClass{abstract=true,parskip=full,fontsize=10pt,
    twoside=semi}{scrartcl}
\LoadClass{scrartcl}
\setparsizes{0pt}{7pt plus 7 pt}{1em plus 1 fill}
\raggedbottom
```

Now the other required packages can be loaded.

```
\RequirePackage{graphicx}
\RequirePackage{babel}
\PassOptionsToPackage{babel}{csquotes}
\bbl@frenchspacing
\RequirePackage{csquotes}

\RequirePackage{geometry}% setup margins
\geometry{paper=a4paper,top=4.8cm, bottom=5.7cm,left=3.8cm,right=4cm,
    head=12.045pt,headsep=\dimexpr1cm-12.045pt\relax}
```

3.3 Bibliography setup

The bibliography will be created using biblatex with biber as backend.

```
\PassOptionsToPackage{backend=biber}{biblatex}
\PassOptionsToPackage{citestyle=authoryear-comp}{biblatex}
\PassOptionsToPackage{bibstyle=apa}{biblatex}
\RequirePackage{biblatex}
```

The APA-style requires a Language mapping. It is set for both supported main languages.

```
\DeclareLanguageMapping{english}{english-apa}
\DeclareLanguageMapping{ngerman}{ngerman-apa}
```

Set up additional Settings to get an equal output, like the Word-version.

```
\defcaptionname{ngerman}{\refname}{Literaturverzeichnis}

\renewcommand*{\nameyeardelim}{\addcomma\space}
```

3.4 Font Choice

The formatting guidelines request Times New Roman as font. Since on most operating systems it is installed as a system font. We check if it is present and the currently used compiler supports the fontspec package.

```
\bool_new:N \g_MUC@fontspec_bool
```

These tests basically use the expl3 methods, but also a fallback for older distributions is implemented to support older distributions. First we implement the compatibility-version.

```
\bool_if:NTF \g_MUC@compatibility_mode {
```

To test if PDFTEX is used, we check if `\pdfmatch` is defined.

```
\expandafter\ifx\csname pdfmatch\endcsname\relax
    \else
      \PassOptionsToPackage{utf8}{inputenc}
      \RequirePackage{inputenc}
\fi
    \bool_gset_false:N \g_MUC@fontspec_bool
```

Next step is the mechanism to check if the used compiler supports fontspec.

```
}{
\sys_if_engine_luatex:TF {
        \bool_gset_true:N \g_MUC@fontspec_bool
    }{
    \sys_if_engine_xetex:TF{
            \bool_gset_true:N \g_MUC@fontspec_bool
    }{
    \sys_if_engine_pdftex:T {
            \bool_gset_false:N \g_MUC@fontspec_bool
    \PassOptionsToPackage{utf8}{inputenc}
    \RequirePackage{inputenc}
        }
        }
    }
}
```

The font fallback is saved in a macro. This prevents the code to be written twice. It is executed when either FONTSPEC is not supported or Times New Roman isn't installed.

```
\cs_new:Nn \MUC@font_Fallback: {
\PassOptionsToPackage{T1}{fontenc}
\RequirePackage{fontenc}
    \IfFileExists{txfonts.sty}{\RequirePackage{txfonts}}{}
}

\if_bool:N \g_MUC@fontspec_bool
\RequirePackage{fontspec}
\fontspec_font_if_exist:nTF {Times~ New~ Roman}
    {\setmainfont{Times~ New~ Roman}}
{\MUC@font_Fallback:}
\else:
\MUC@font_Fallback:
\fi:
```

3.5 PDF/A-mode and metadata creation

The PDF archiving mode is created using the pdfx package. For the correct creation the PDF/A format and the `.xmpdata` file for metadata specific system settings may be required. Therefore this mode is only active if the `forcefinal` mode is enabled.

```
\if_bool:N \g_MUC@pdfa_bool
    \PassOptionsToPackage{a-1b}{pdfx}
    \RequirePackage{pdfx}
    \hypersetup{hidelinks}
    \iow_new:N \MUC@xmpdata_stream
    \iow_open:Nn \MUC@xmpdata_stream {\jobname.xmpdata}
    \tl_new:N \g_MUC@xmp_title
```

```
    \tl_new:N \g_MUC@xmp_author

    \newcommand*{\MUC@passTitleData}{
    \begingroup
    \def\newline{}
    \def\\{}
    \xdef\MUC@author{\@author}
    \xdef\MUC@title{\@title}
    \cs_set:Npn \and {\exp_not:n {\exp_not:N \sep}}
    \tl_gset:Nx \g_MUC@xmp_title {\@title}
    \tl_gset:Nx \g_MUC@xmp_author {\@author}
    \endgroup
    \iow_now:Nx \MUC@xmpdata_stream
     {
     \exp_not:N \Title{\tl_use:N \g_MUC@xmp_title}
     ^^J
     \exp_not:N \Author{\tl_use:N \g_MUC@xmp_author}
     ^^J
     \exp_not:N \Org{Mensch~ und~ Computer~ 2017}
     }
    }
```

If no PDF/A shall be created, the `\maketitle` data will only be passed to the headline.

```
\else:
    \msg_new:nnn {mucproc} {no-pdfa}{The~ mucproc~ class~ won't~ create~ PDF/A-mode.}
    \msg_info:nn {mucproc} {no-pdfa}
    \newcommand*{\MUC@passTitleData}{
        \begingroup
     \def\newline{}
     \def\\{}
     \xdef\MUC@author{\@author}
     \xdef\MUC@title{\@title}
        \endgroup
    }
\fi:
```

4 Title creation

The title creation process also invokes the mechanism to pass the title data to headline and metadata. It can be initialized calling `\maketitle`.

4.1 Institution footnotes

The MuC formatting guidelines request the institutions typeset similar to footnotes, though they shouldn't be placed at the bottom of the page. The standard footnote mechanism is quite complicated, though we use the LaTeX3 clist datatype to create a special mechanism implementing these

"footnotes". The mechanism is called inside `\maketitle` but created outside of that macro definition to leave the code more readable.

All Arguments of the `\thanks` command will be saved in the comma list called `\l_MUC@thanks_clist`.

```
\clist_new:N \l_MUC@thanks_clist
```

The value of the footnote counter has to be expanded when the `\thanks` macro is called. Otherwise the footnotemarks might not match. To achieve this expansion order the `\MUC@thanks:nn` command takes the counter value as first argument and the institution name as second.

```
\cs_set:Nn \MUC@thanks:nn {
    \clist_gput_right:Nn \l_MUC@thanks_clist
        {{#2\def\@thefnmark{#1}\@makefnmark}
    }}
```

Inside the real `\thanks` macro the first argument will be expanded directly. Additionally `\MUC@thanks` sets its own first (optional) argument as a label for the property list `\g_MUC@InstRef_prof`. This allows the footnote counter value to be accessed through the label.

```
\bool_if:NTF \g_MUC@compatibility_mode
{\NewDocumentCommand{\MUC@thanks}{om}{
    \footnotemark%
    \exp_args:Nx \MUC@thanks:nn {\thefootnote}{#2}
    }
}{
\NewDocumentCommand{\MUC@thanks}{om}{
    \footnotemark%
    \IfNoValueF{#1}{
        \prop_if_exist:NF \g_MUC@InstRef_prop {
            \prop_new:N \g_MUC@InstRef_prop}
        \prop_put:Nnx \g_MUC@InstRef_prop {#1} {\thefootnote}
        }
    \exp_args:Nx \MUC@thanks:nn {\thefootnote}{#2}
    }
}
```

The mechanism seems to work like the typical LaTeX label/ref though no backref is supported. To inform the user about this an error will be thrown when a label is accessed which does not exist.

```
\bool_if:NTF \g_MUC@compatibility_mode {
\msg_new:nnn {mucproc} {thanksref-unavailable} {
    Compatibilty~ mode~ enabled.\\\\
    The~ institution~ reference~ mechanism~ is~ not available.\\\\
    Update~ Your~ TeX~ distribution~ to~ be~ able~
        to~ use~ this~ feature.
    }
\msg_warning:nn {mucproc} {thanksref-unavailable}
    \newcommand\MUC@thanksref[1]{$\dagger$}
}{
```

```
\msg_new:nnn {mucproc} {thanks-label-missing} {
    No~ institute~ labeled~ '#1'.\\\\
    Add~ a~ label~ using~ the~ optional~ argument~
        of~ the~ \thanks~ command.
    }

\NewDocumentCommand\MUC@thanksref{m}{
    \prop_if_in:NnTF \g_MUC@InstRef_prop {#1} {
    \def\@thefnmark{\prop_item:Nn \g_MUC@InstRef_prop {#1}}%
    \@makefnmark
    }
    {\msg_error:nn {mucproc} {thanks-label-missing}}
}
}
```

After the title has been set, the institutional notes shall be typeset.

```
\newcommand*\typeset@MUC@thanks {
    \clist_use:Nn \l_MUC@thanks_clist {\rule{0pt}{0pt}\\}}
```

4.2 Typesetting the title

As written before, `\maketitle` has to execute additional jobs. Since the title fields also can contain other macros or even line breaks, the expansion process has to be controlled carefully.

```
\renewcommand*{\maketitle}{
\begingroup
```

Before passing the data to the headlines or metadata using the earlier defined `\MUC@passTitleData` we need some redefinitions for the commands, which can be used inside the title fields.

```
\DeclareExpandableDocumentCommand{\thanks}{om}{}
\def\thanksref##1{}
\def\and{\unskip,~\ignorespaces}
\MUC@passTitleData
```

Not the actual title creation begins redefining the user macros to those prepared for that purpose.

```
\let\titlepage@restore\relax
\renewcommand*\thefootnote{\@arabic\c@footnote}
\let\thanks\MUC@thanks
\let\thanksref\MUC@thanksref
```

Now only the title fields have to be typeset.

```
\global\@topnum=\z@
\setparsizes{\z@}{\z@}{\z@\@plus 1fil}\par@updaterelative
\null
\vskip 41pt
{\usekomafont{title}{\Huge \@title \par}}%
    {\ifx\@subtitle\@empty\else\vskip .5em\usekomafont{subtitle}%
    \Large\@subtitle\par\fi}%
```

```
    \vskip 7pt
{%
\usekomafont{author}{%
\lineskip .5em%
\@author
\par
}%
}%
\par
\vskip 8pt
\bool_if:NTF \g_MUC@compatibility_mode
    {\thispagestyle{empty}}
    {\thispagestyle{MUC@titlepagestyle}}
\typeset@MUC@thanks
\par
\vspace{8pt}
\endgroup
\setcounter{footnote}{0}%
\global\let\and\relax
\global\let\thanksref\relax
}
```

4.3 Anonymization and additional author information

The anonymization is realized checking the setting using the corresponding if-expression. The `\@author` information to the language dependent meaning of `\MUCanonymousauthor`.

```
\newcaptionname{ngerman}{\MUCanonymousauthor}{Anonymisierte Fassung}
\newcaptionname{english}{\MUCanonymousauthor}{anonymous version}
\ifMUC@anonymous
    \def\@author{\MUCanonymousauthor}
    \def\author#1\relax
```

The `authoraddendum` environment shall create no output, if the anonymous mode is active.

```
    \RequirePackage{comment}
    \excludecomment{authoraddendum}
\else
```

`authoraddendum` The Implementation works via the typical xparse mechanisms.

```
    \NewDocumentEnvironment{authoraddendum}{d<>om}{
        \par
        \IfNoValueTF{#2}{
        {\usekomafont{authoraddendum}#3}\par\medskip
        }{
        \begin{minipage}[t]{3cm}
        \vspace{0pt}
        \IfNoValueTF{#1}{
            \includegraphics[width=\linewidth]{#2}
```

```
        }{
            \includegraphics[#1]{#2}
        }
        \end{minipage}\hspace{.5cm}%
        \begin{minipage}[t]{\dimexpr\linewidth-3.5cm\relax}
        \vspace{0pt}
        {\usekomafont{authoraddendum}#3}\par\medskip%
        }
        \usekomafont{authoraddendum}
        }{
        \IfNoValueF{#2}{\end{minipage}}
        \par
        }
\fi
```

4.4 Visible adjustment of sizes and spacing

To match the Word template we have to adjust a bunch of spacing settings as well as font sizes. The easiest way to achieve this is using the KOMA-Script mechanisms.

4.5 Structure Settings

Word indents the section headings independent from the width of the section number. To achieve this a macro called `\MUC@sectionnumberbox` is created and all section formats are redefined. In compatibility mode these settings will be ignored.

```
\reverse_if:N  \if_bool:N \g_MUC@compatibility_mode
\newcommand*{\MUC@sectionnumberbox}[1]{\makebox[28bp][l]{#1\autodot}}
\renewcommand*{\sectionformat}{
    \MUC@sectionnumberbox{\thesection}}
\renewcommand*{\subsectionformat}{
    \MUC@sectionnumberbox{\thesubsection}}
\renewcommand*{\subsubsectionformat}{
    \MUC@sectionnumberbox{\thesubsubsection}}
```

The spacing around the headings also has to be adjusted:

```
\RedeclareSectionCommand[afterskip=1.5ex plus .2ex,%
    beforeskip=-3.6ex plus -1ex minus -.2ex]{section}
\RedeclareSectionCommand[afterskip=.5ex plus .2ex,%
    beforeskip=-1ex plus -1ex minus -.2ex]{subsection}
\RedeclareSectionCommand[afterskip=.5ex plus .2ex,%
    beforeskip=-1ex plus -1ex minus -.2ex]{subsubsection}
\RedeclareSectionCommand[afterskip=1em,%
    beforeskip=1ex plus 1ex minus .2ex]{paragraph}
\RedeclareSectionCommand[afterskip=1em,%
    beforeskip=1ex plus 1ex minus .2ex]{subparagraph}
\fi:
\ExplSyntaxOff
```

The font settings will be done together with all other font settings in the next subsection.

4.6 Font Settings

Adjusting the font settings to match the Word version:

```
\setkomafont{disposition}{}
\setkomafont{title}{}
\setkomafont{author}{\large}
\setkomafont{section}{\LARGE}
\setkomafont{subsection}{\Large}
\setkomafont{caption}{\footnotesize\itshape}
\setkomafont{captionlabel}{\usekomafont{caption}}
\newkomafont{abstractcontent}{\small}
\newkomafont{abstract}{\normalsize\bfseries}
\newkomafont{authoraddendum}{\small}
```

4.7 Abstract

The `abstract` environment is redefined to match the Word version.

```
\renewenvironment{abstract}{%
    \setparsizes{0pt}{4pt}{0pt plus 1 fill}{%
    \usekomafont{abstract}%
    \abstractname}\par%
    \usekomafont{abstractcontent}}{}
```

4.8 Pagestyle

The demo document shows a headline containing author and document title. To achieve this that information was passed inside the `\maketitle` macro and saved in the `\MUC@author` and `\MUC@title` macros. There is also a possibility to set a copyright-information box similar to the Word template.

```
\ifMUC@pagestyle
\PassOptionsToPackage{headsepline}{scrlayer-scrpage}
    \RequirePackage{scrlayer-scrpage}
    \ifMUC@copyrightinfo
```

Nowadays it is possible to write the complete content for the scrlayers directly inside the definition. To support older distributions as well the content is wrapped into a box. The most absolute length values have been extracted from different measurements to match the Word version.

```
        \newsavebox\MUC@copyrightbox
     \begin{lrbox}{\MUC@copyrightbox}
     \setlength{\fboxrule}{.7bp}%
     \setlength{\fboxsep}{6.5pt}%
     \fbox{\hspace*{.8pt}\parbox[t][\dimexpr2cm-2\fboxsep
```

```
            -2\fboxrule\relax][t]{%
                \dimexpr\textwidth-2\fboxsep-2\fboxrule%
                -1.6pt\relax}{%
     Platzhalter für DOI und ggf. Copyright Text.
        (Bitte nicht entfernen).\par
     \vspace{7pt}
     Name, Vorname (2017): Titel.%
     Tagungsband Mensch und Computer 2017.
        Gesellschaft für Informatik.%
     DOI: xxxxxx}%
     }
     \end{lrbox}
    \DeclareNewLayer[align=tl,area={\dimexpr 1in
         + \hoffset + \oddsidemargin\relax}{%
        \dimexpr\paperheight-3.9cm\relax}{\textwidth}{2cm},
        contents={\usebox\MUC@copyrightbox}]{copyright}
    \DeclareNewPageStyleByLayers{MUC@titlepagestyle}{copyright}
    \else
    \DeclareNewPageStyleAlias{MUC@titlepagestyle}{empty}
    \fi
    \clearpairofpagestyles
    \ifMUC@headline
        \lohead{\MUC@title}
        \rehead{\csname MUC@author\endcsname}
    \fi
    \setkomafont{pagehead}{\normalfont}
    \setkomafont{pagefoot}{}
\fi
```