

# The papermas package\*

H.-Martin Münch  
(Martin dot Muench at Uni-Bonn dot de)

2010/06/03

## Abstract

This package allows to compute the number of sheets of paper needed to print a document as well as the mass of that printed version of the document, usefull e.g. when sending it by mail to determine the postage.  
(The number of pages of a document can be determined with the `pagesLTS` package.) –  
Further this package allows to compute “*base* to the power of *exponent*” inside `LATEX`.

Disclaimer for web links: The author is not responsible for any contents referred to in this work unless he has full knowledge of illegal contents. If any damage occurs by the use of information presented there, only the author of the respective pages might be liable, not the one who has referred to these pages.

Save per page about 200 ml water, 2 g CO<sub>2</sub> and 2 g wood:  
Therefore please print only if this is really necessary.

---

\*This file has version number v1.0b, last revised 2010/06/03, documentation dated 2010/06/03.

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Usage</b>	<b>3</b>
2.1	Options	3
2.1.1	format	3
2.1.2	masss	3
2.1.3	pagespersheet	3
<b>3</b>	<b>Alternatives</b>	<b>3</b>
<b>4</b>	<b>Example</b>	<b>4</b>
<b>5</b>	<b>The implementation</b>	<b>6</b>
<b>6</b>	<b>Installation</b>	<b>13</b>
6.1	Downloads	13
6.2	Package, unpacking TDS	14
6.3	Refresh file name databases	14
6.4	Some details for the interested	14
6.5	Compiling the example	15
<b>7</b>	<b>Things suggested to be done</b>	<b>15</b>
<b>8</b>	<b>Acknowledgements</b>	<b>15</b>
<b>9</b>	<b>History</b>	<b>15</b>
	[2010/06/01 v1.0]	15
	[2010/06/03 v1.0b]	15
<b>10</b>	<b>Index</b>	<b>16</b>

## 1 Introduction

This package is kind of an add-on to my `pagesLTS` package, but because that already uses some resources and computing the number of sheets of paper or the paper mass probably is not needed so often, this was made into a separate package.

It allows to compute the number of sheets of paper needed to print a document (usefull when the paper is running out) as well as the mass of that printed version of the document, usefull e. g. when sending it by mail to determine the postage.

**Warning/Disclaimer:** The mass of (printer's) ink has to be added (and that of envelope, address sticker, stamps,...). So, this is only an estimation without guarantee – do not sue me, if you have got to pay excess postage!

Further this package allows to compute “*base* to the power of *exponent*” inside  $\text{\LaTeX}$ .

The name `papermas` is short for paper mass but written with only one `s`, because some software has problems with names with more than eight letters.

It is `mass` and gives a result in grammes [g], because the weight  $F = m \cdot g$  (really  $\vec{F} = m \cdot \vec{g}$ ) [N] would need the knowledge about the gravitational acceleration  $g$  (depending on place and time, in central Europe approximately  $9.81 \text{ m/s}^2$ ) and give a result in NEWTON, which probably is not very usefull.

## 2 Usage

Just load the package placing

```
\usepackage[<options>]{papermas}
```

in the preamble of your L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> source file (preferably after calling the `pagesLTS` package).

Because the `pagesLTS` package is used to get the total number of pages, please place a `\pagenumbering{...}` with appropriate argument (e.g. `arabic`, `roman`, `Roman`, `fnsymbol`, `alph`, or `Alph`) right behind `\begin{document}` (see documentation of `pagesLTS` package).

Now you can say

```
This document consists of $\arabic{pagesLTS.pagenr}$~pages.
When printing $\papermaspagespersheet$~pages on one sheet of
paper, $\papermasssheets$~sheets will be needed. For
ISO~A~\papermasformat\ paper of $\papermasssss \unit{g}\unit{m}^{-2}$
specific mass, the printout will have a mass of about
$\papermasstotal \unit{g}$.
```

to get e.g.

```
This document consists of 101 pages. When printing 4 pages on one
sheet of paper, 26 sheets will be needed. For ISO A 4 paper of 80 g m-2
specific mass, the printout will have a mass of about 130 g.
```

### 2.1 Options

**options**      The `papermas` package takes the following options:

#### 2.1.1 format

**format**      Option `format` wants to know the ISO A...format of the paper used for printing, i.e. `format=4` means ISO A4 paper format (which is also the default).

#### 2.1.2 masss

**masss**      Option `masss` wants to know the specific (therefore the third s) mass of the paper used for printing in g/m<sup>2</sup>. The default is `masss=80`, i.e. 80 g/m<sup>2</sup>.

#### 2.1.3 pagespersheet

**pagespersheet**      Option `pagespersheet` wants to know, how many pages are to be printed on one sheet of paper. `pagespersheet=2` could mean duplex printing or printing two pages on one side of paper while keeping the back side blank. This does not influence the real printing process! So, if this number differs from the one chosen for printing, the result will be wrong, of course.

## 3 Alternatives

For determining the number of pages (not sheets of paper) instead of the `pagesLTS` package the alternatives listed in the description of that package could be used, but then the according code in this package would need to be changed (and also e.g. the `ifcounter` used here).

With the `totpages` package optionally the number of sheets of paper needed to print the document can be computed, too.  
(See `pagesLTS` documentation.)

(You programmed or found another alternative, which is available at [CTAN](#)? OK, send an e-mail to me with the name, location at [CTAN](#), and a short notice, and I will probably include it in the list above.)

About how to get those packages, please see subsection [6.1](#).

## 4 Example

```
1 (*example)
2 \documentclass[british,a4paper]{article}
3 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4 \usepackage[hyperref]{2010/05/13}% v6.81e
5 \hypersetup{
6   extension=pdf,%
7   plainpages=false,%
8   pdfpagelabels=true,%
9   hyperindex=false,%
10  pdflang={en},%
11  pdftitle={papermas package example},%
12  pdfauthor={Hans-Martin Muench},%
13  pdfsubject={Example for the papermas package},%
14  pdfkeywords={LaTeX, papermas, Hans-Martin Muench},%
15  pdfview=Fit,%
16  pdfstartview=Fit,%
17  pdfpagelayout=SinglePage,%
18  bookmarksopen=false%
19 }
20 \usepackage[alphalph]{2010/04/18}% v2.3
21 \usepackage[pagecontinue=true,alphMult=ab,AlphMulti=AB,fnsymbolmult=true,romanMult=true,RomanMult=true]{papermas}
22 %% These are the default options. %%
23 \usepackage[format=4,masss=80,pagespersheet=2]{papermas}
24 %% These are the default options. %%
25 \listfiles
26 \begin{document}
27 \pagenumbering{arabic}
28
29 \section*{Example for papermas}
30 \markboth{Example for papermas}{Example for papermas}
31
32 This example demonstrates the use of package\newline
33 \textsf{papermas}, v1.0b as of 2010/06/03 (HMM).\newline
34 The used options were \texttt{format=4} (ISO~A4),
35 \texttt{masss=80} ( $\text{unit}\{\text{g}\}\text{unit}\{\text{m}\}^{-2}$ ), and\newline
36 \texttt{pagespersheet=2} (pages per sheet of paper,
37 i.\,e. either duplex printing or\newline
38 printing two pages on one side of a sheet of paper with blank back side).\newline
39 (These are the default options.)\newline
40 For more details please see the documentation!\newline
41
42 \bigskip
43
44 This document consists of
45 \lastpageref{LastPages}~(\arabic{pagesLTS.pagenr})~pages.
46 When printing  $\text{papermaspagespersheet}$ ~pages on one sheet of
47 paper,  $\text{papermasssheets}$ ~sheets will be needed. For
48 ISO~A~\papermasformat~paper of  $\text{papermasmasss}$   $\text{unit}\{\text{g}\}\text{unit}\{\text{m}\}^{-2}$ 
49 specific mass, the printout will have a mass of about
50  $\text{papermasstotal}$   $\text{unit}\{\text{g}\}$ $.
51
52 \bigskip
53
54 Save per page about  $200\text{unit}\{\text{ml}\}$ ~water,  $2\text{unit}\{\text{g}\}$ ~CO $_2$ ~$
55 and  $2\text{unit}\{\text{g}\}$ ~wood:\newline
56 Therefore please print only if this is really necessary.\newline
57 I do NOT think, that it is necessary to print THIS file, really\newline
58 (at least not after this page)!
59
```

```

60
61 \newpage Page \thepage
62 \newpage Page \thepage
63 \newpage Page \thepage
64 \newpage Page \thepage
65 \newpage Page \thepage
66 \newpage Page \thepage
67 \newpage Page \thepage
68 \newpage Page \thepage
69 \newpage Page \thepage
70 \newpage Page \thepage
71 \newpage Page \thepage
72 \newpage Page \thepage
73 \newpage Page \thepage
74 \newpage Page \thepage
75 \newpage Page \thepage
76 \newpage Page \thepage
77 \newpage Page \thepage
78 \newpage Page \thepage
79 \newpage Page \thepage
80 \newpage Page \thepage
81 \newpage Page \thepage
82 \newpage Page \thepage
83 \newpage Page \thepage
84 \newpage Page \thepage
85 \newpage Page \thepage
86 \newpage Page \thepage
87 \newpage Page \thepage
88 \newpage Page \thepage
89 \newpage Page \thepage
90 \newpage Page \thepage
91 \newpage Page \thepage
92 \newpage Page \thepage
93 \newpage Page \thepage
94 \newpage Page \thepage
95 \newpage Page \thepage
96 \newpage Page \thepage
97 \newpage Page \thepage
98 \newpage Page \thepage
99 \newpage Page \thepage
100 \newpage Page \thepage
101 \newpage Page \thepage
102 \newpage Page \thepage
103 \newpage Page \thepage
104 \newpage Page \thepage
105 \newpage Page \thepage
106 \newpage Page \thepage
107 \newpage Page \thepage
108 \newpage Page \thepage
109 \newpage Page \thepage
110 \newpage Page \thepage
111 \newpage Page \thepage
112 \newpage Last page \thepage.
113 \end{document}
114 </example>

```

## 5 The implementation

We start off by checking that we are loading into L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> and announcing the name and version of this package.

```
115 \*package)
116 \NeedsTeXFormat{LaTeX2e}[1994/06/01]
117 \ProvidesPackage{papermas}[2010/06/03 v1.0b
118           Computes paper mass of a printout (HMM)]%
119
```

A short description of the papermas package:

```
120 %% Allows to compute the number of sheets of paper
121 %% needed to print a document as well as the
122 %% mass of that printed version of the document,
123 %% usefull e. g. when sending it by mail to determine the postage.
124 %% Warning/Disclaimer: Mass of (printer's) ink has to be added
125 %% (and that of envelope, address sticker, stamps,...)!
126 %% So, this is only an estimation without guarantee -
127 %% do not sue me, if you have got to pay excess postage!
128 %% Further this package allows to compute
129 %% "base to the power of exponent" inside TeX.
130
```

For the handling of the options we need the kvoptions package of Heiko Oberdiek (see subsection 6.1):

```
131 \RequirePackage{kvoptions}[2010/02/22]% v3.7
```

For the total number of pages we need the pagesLTS package of myself (see subsection 6.1):

```
132 \RequirePackage{pagesLTS}[2010/06/03]% v1.1b
```

A last information for the user(s):

```
133 %% papermas may work with earlier versions of those packages,
134 %% but this was not tested. Please consider updating your packages
135 %% to the most recent version (if they are not already the most
136 %% recent version).
137
```

See subsection 6.1 about how to get them.

The options are introduced:

```
138 \SetupKeyvalOptions{family = papermas,prefix = papermas@}
139 \DeclareStringOption[4]{format}[4]%      paper foormat, ISO A...,
140                                     %      default: (ISO A) 4
141 \DeclareStringOption[80]{masss}[80]%      specific mass of the paper,
142                                     %      default: 80 (g/(m^2))
143 \DeclareStringOption[2]{pagespersheet}[2]% number of pages per sheet,
144                                     % for duplex printing this is 2.
145
146 \ProcessKeyvalOptions*
147
```

**unit** We want to use the \unit command from *Scientific WorkPlace*, therefore we need to copy the definition:

```
148 %% Code from tcilatex.tex, Macros for Scientific Word and Scientific WorkPlace 5.5 <06 Oct 20
149 %% Copyright (C) 2005 Mackichan Software, Inc.
150 %% That macro file is NOT proprietary and may be freely copied and distributed.
151 %% \def was changed to \gdef
152 \gdef\unit#1{\mathord{\thinspace\rm #1}}%
153 %% End of code from tcilatex.tex
154
```

Even if diverse commands are not defined yet, we do not want a  
 $\LaTeX$  Error: ... undefined.

```

155 \ifx\papermasstotal\undefined \gdef\papermasstotal{\textbf{??}}%
156 \fi
157 \ifx\papermasstotal\undefined \gdef\papermasstotal{\textbf{??}}%
158 \fi
159 \ifx\papermasformat\undefined \gdef\papermasformat{\textbf{??}}%
160 \fi
161 \ifx\papermasssss\undefined \gdef\papermasssss{\textbf{??}}%
162 \fi
163 \ifx\papermaspagespersheet\undefined \gdef\papermaspagespersheet{\textbf{??}}%
164 \fi
165 \ifx\papermasssheets\undefined \gdef\papermasssheets{\textbf{??}}%
166 \fi
167

```

Introducing some new counters:

```

168 \newcounter{papermas@rerun}
169 \newcounter{papermas@base}
170 \newcounter{papermas@exp}
171 \newcounter{papermas@result}
172 \newcounter{papermas@ini}
173 \setcounter{papermas@ini}{1}
174

```

$\backslash$ papermas@powerof We need a command to compute “base to the power of exponent” ( $base^{exponent}$ ).  
(Is this really not already implementet in  $\LaTeX$ ?!)

```

175 \newcommand\papermas@powerof[2]{%
176   \setcounter{papermas@base}{\#1}
177   \setcounter{papermas@exp}{\#2}
178   \ifnum \value{papermas@ini}=1
179     \setcounter{papermas@result}{\value{papermas@base}}
180     \setcounter{papermas@ini}{0}
181   \ifnum \value{papermas@exp}=0%
182     \setcounter{papermas@result}{1}
183   \else
184     \addtocounter{papermas@exp}{-1}
185   \fi
186 \fi
187 \ifnum \value{papermas@exp}=0%
188   \setcounter{papermas@ini}{1}
189 \else
190   \multiply \value{papermas@result} \value{papermas@base}
191   \addtocounter{papermas@exp}{-1}
192   \papermas@powerof{\#1}{\value{papermas@exp}}
193 \fi%
194 }
195

```

$\backslash$ papermas@totmass This is the internal command, which computes the total paper mass of the printed document.

```

196 \newcommand\papermas@totmass{%
197   \newcounter{papermasA}% paper mass for ISO A...
198   \setcounter{papermasA}{\papermas@format}% e. g. 4

```

We check whether papermasA has a resonable value:

```

199   \ifnum \value{papermasA}<0%
200     \PackageError{papermas}{Option format has no valid value}%
201     {The format option of the papermas package\MessageBreak%
202      only takes whole, non-negative numbers (0, 1, 2, 3,...),\MessageBreak%
203      because this should be the paper format\MessageBreak%
204      ISO A 0, 1, 2, 3,...\MessageBreak%
205      Found instead: \papermas@format \MessageBreak%
206     }
207   \else%

```

`papermasA` has a reasonable value. We introduce a new counter `papermasmasss` and initialize it with the value given in option `masss`, i.e. `\papermas@masss`.

```
208 \newcounter{papermasmasss}% always 0
209 \setcounter{papermasmasss}{\papermas@masss}% default: 80
```

Counters are integers, but the amount of the mass of a single sheet of paper in most cases is not an integer, therefore we multiply with 100 to get two digits behind the decimal separator.

(Later we need to divide by 100 again, of course.)

```
210 \multiply \value{papermasmasss} 100 % default: 8000
```

We check whether `papermasmasss` has a reasonable value, i.e.  $> 0$ :

```
211 \ifnum \value{papermasmasss}<1%
212 \PackageError{pagesLTS}{Option masss has no valid value}%
213 {The masss option of the papermas package\MessageBreak%
214 only takes positive numbers,\MessageBreak%
215 because this should be the mass per square meter\MessageBreak%
216 of a single sheet of your paper.\MessageBreak%
217 Found instead: \papermas@masss \MessageBreak%
218 }
219 \else
```

`masss` has a reasonable value, and therefore also `\papermas@masss` and `papermasmasss`.

We check whether option `pagesperssheet` has a reasonable value, i.e.  $\geq 1$ :

```
220 \newcounter{papermasPPS}% is 0
221 \setcounter{papermasPPS}{\papermas@pagesperssheet}% default 2
222 \ifnum \value{papermasPPS} < 1%
223 \PackageError{papermas}{%
224 The number of pages per sheet must be positive.}%
225 You cannot print less than one TeX page per sheet of paper.\MessageBreak%
226 The value found was \papermas@pagesperssheet .\ \MessageBreak%
227 }
228 \else
```

`pagesperssheet` has a reasonable value, and therefore also `\papermas@pagesperssheet` and `papermasTmpA`.

We introduce a new counter `papermas@sheets` for the number of sheets printed and initialize it with the number of pages as computed by package `pagesLTS`, i.e. `pagesLTS.pagenr`.

```
229 \newcounter{papermas@sheets}
230 \setcounter{papermas@sheets}{\arabic{pagesLTS.pagenr}}%
```

When more than one page is printed on one sheet of paper, the number of sheets needed for printing is decreased:

```
231 \divide \value{papermas@sheets} by \value{papermasPPS}%
```

`\divide` cuts off all digits behind the decimal separator, but if there are digits  $> 0$ , this means that there is an additional, last sheet, which is only partially covered with print (e.g. only one side of it for duplex printing an odd number of pages). In that case, we have to add one sheet of paper to the number of sheets needed.

```
232 \newcounter{papermas@tmpn}
233 \setcounter{papermas@tmpn}{\arabic{papermas@sheets}}%
234 \multiply \value{papermas@tmpn} \value{papermasPPS}%
235 \ifnum \value{papermas@tmpn}=\value{pagesLTS.pagenr}
236 \relax
237 \else
238 \addtocounter{papermas@sheets}{1}%
239 \fi
```

Now we can multiply the specific mass of 100 sheets with the number of sheets needed for printing:

```
240 \multiply \value{papermasmasss} \value{papermas@sheets}
241 % default: 8000
```



The result is in  $\text{g m}^{-2}$ .

A sheet with format ISO A0 has a size of  $1 \text{ m}^2$ ,  
a sheet with format ISO A1 has a size of  $1 \text{ m}^2 \cdot 2^{-1}$ ,  
a sheet with format ISO A2 has a size of  $1 \text{ m}^2 \cdot 2^{-2}$ ,  
... a sheet with format ISO An has a size of  $1 \text{ m}^2 \cdot 2^{-n}$ .  
Therefore we compute  $2^{\text{\value{papermasA}}}$ :

```
242 \papermas@powerof{2}{\value{papermasA}}
```

The result is saved in `papermas@result`.

We divide the specific paper mass by `papermas@result`:

```
243 \divide \value{papermasmasss} by \value{papermas@result}
244 % default: 16000 / 2^{\value{papermasA}}
```

We need to get the division by 100 and the digits after the decimal separator right:

```
245 % for the example 297 is used
246 \newcounter{papermas@tmpm}
247 \setcounter{papermas@tmpm}{\arabic{papermasmasss}}% m:297 n: o: p: q:
248 \setcounter{papermas@tmpn}{\arabic{papermasmasss}}% m:291 n:291 o: p: q:
249 \divide \value{papermas@tmpn} by 100% m:297 n:2 o: p: q:
250 \newcounter{papermas@tmpo}
251 \setcounter{papermas@tmpo}{\arabic{papermas@tmpn}}% m:291 n:2 o:2 p: q:
252 \multiply \value{papermas@tmpn} 10% m:297 n:20 o:2 p: q:
253 \divide \value{papermas@tmpm} by 10% m:29 n:20 o:2 p: q:
254 \newcounter{papermas@tmpp}
255 \setcounter{papermas@tmpp}{\arabic{papermas@tmpm}}
256 \addtocounter{papermas@tmpp}{-\arabic{papermas@tmpn}}%m:29 n:20 o:2 p:9 q:
257 % 29 - 20 = 9
258 \multiply \value{papermas@tmpp} 10% m:290 n:20 o:2 p:9 q:
259 \newcounter{papermas@tmpq}
260 \setcounter{papermas@tmpq}{\arabic{papermasmasss}}
261 \addtocounter{papermas@tmpq}{-\arabic{papermas@tmpm}}%m:290 n:20 o:2 p:9 q:7
262 % 297 - 290 = 7
```

Now rounding mathematically correct, i. e.  $\geq 0.5$  becomes 1 (and remember a possible amount carried forward!) and  $< 0.5$  becomes 0.

```
263 \ifnum\value{papermas@tmpq}>4
264 \addtocounter{papermas@tmpp}{1}% m:290 n:20 o:2 p:10 q:7
265 \ifnum\value{papermas@tmpp}>9% m:290 n:20 o:2 p:10 q:7
266 \addtocounter{papermas@tmpp}{1}% m:290 n:20 o:3 p:10 q:7
267 \setcounter{papermas@tmpp}{0}% m:290 n:20 o:3 p:0 q:7
268 \fi
269 \fi
```

The result in the example above is  $297/100 = 2.97 \approx 3.0$ . We write this into `\papermastmpr` and the options' values into temporary definitions, as well as the number of sheets:

```
270 \edef\papermastmpr{\arabic{papermas@tmpo}.\arabic{papermas@tmpp}}%
271 \edef\papermastmpformat{\papermas@format}%
272 \edef\papermastmpmasss{\papermas@masss}%
273 \edef\papermastmppagesperssheet{\papermas@pagesperssheet}%
274 \edef\papermastmpt{\arabic{papermas@sheets}}%
```

We use the `pagesLTS` package, which already was used to determine the total number of pages, to check for the counter `papermassttl`. If it exists, nothing is done, if it does not exist, it is declared as `\newcounter` (and by default set to zero).

```
275 \pagesLTS@ifcounter{papermassttl}
```

If the `papermassttl` counter value already has the value of `papermasmasss`, everything is fine.

```
276 \ifnum\value{papermassttl}=\value{papermasmasss}
277 \relax
```

Otherwise we need another run of  $\text{\LaTeX}$ .

```
278 \else
```

```

279         \PackageWarningNoLine{papermas}{%
280             Number of pages may have changed.\MessageBreak%
281             Rerun to get it right.\MessageBreak%
282         }%
283     \fi

```

In any case, we set the counter `papermassttl` to the current value of `papermasssss`.

```

284     \setcounter{papermassttl}{\arabic{papermasssss}}

```

Because we want to write out into the aux-file, we need the expanded value (as string) of `papermasssss`:

```

285     \edef\papermastmps{\arabic{papermasssss}}%

```

If we are allowed to write into the aux-file, we do it here. If we are not allowed to do it, the `pagesLTS` package already gave an according error message.

```

286     \if@filesw%

```

When it is read from the aux-file and when its content is processed, the counter `papermassttl` might not have been defined yet. Therefore we again use the `\pagesLTS@ifcounter` command of the `pagesLTS` package.

```

287         \immediate\write\@auxout{\string
288         \pagesLTS@ifcounter{papermassttl}}%

```

We set the counter `papermassttl` to the value `\papermastmps`, i. e. `\arabic{papermasssss}`. In the next compilation run, it will be checked, whether `\value{papermassttl}=\value{papermasssss}` (see above). If this is the case, everything is OK, no changes happened, and no rerun is necessary (at least not for `papermas`).

```

289         \immediate\write\@auxout{\string
290         \setcounter{papermassttl}{\papermastmps}}%

```

What we do need, is to get the determined `\papermastmpr` to the user. Therefore

1. we define `\papermassttotal` in the aux-file, where the user can look it up
2. we define `\papermassttotal`, so the user can e. g. write

```

This document consists of $\arabic{pagesLTS.pagenr}$~pages.
When printing $\papermaspagesperssheet$~pages on one sheet of
paper, $\papermasssheets$~sheets will be needed. For
ISO~A~\papermasformat\ paper of $\papermasssss\unit{g}\unit{m}^{-2}$
specific mass, the printout will have a mass of about
$\papermassttotal\unit{g}$.

```

```

291     \immediate\write\@auxout{\string
292     \gdef\string\papermassttotal{\papermastmpr}}%
293     \immediate\write\@auxout{\string
294     \gdef\string\papermasformat{\papermastmpformat}}%
295     \immediate\write\@auxout{\string
296     \gdef\string\papermasssss{\papermastmpmasss}}%
297     \immediate\write\@auxout{\string
298     \gdef\string\papermaspagesperssheet{\papermastmppagesperssheet}}%

```

3. we give at the screen the information about the `\papermassttotal` (see `\AtVeryEnd` below)
4. which will also appear in the log-file.

We want to give also `\papermastmpt = \arabic{papermas@sheets}` to the user, i. e. the number of sheets needed to print the document. Therefore we follow the same procedure:

```

299     \immediate\write\@auxout{\string
300     \gdef\string\papermasssheets{\papermastmpt}}%
301     \fi%

```

```

302     \fi%
303     \fi%
304 \fi%
305 }
306

```

`\AtBeginDocument` `\AtBeginDocument` it is checked whether some commands, which are/will be defined via the `aux`-file, are undefined yet:

```

307 \AtBeginDocument{%
308   \def\papermas@undefined{\textbf{??}}
309   \setcounter{papermas@rerun}{0}
310   \ifx\papermasstotal\papermas@undefined \addtocounter{papermas@rerun}{000001} \fi
311   \ifx\papermasstotal\papermas@undefined \addtocounter{papermas@rerun}{000010} \fi
312   \ifx\papermasformat\papermas@undefined \addtocounter{papermas@rerun}{000100} \fi
313   \ifx\papermasmasss\papermas@undefined \addtocounter{papermas@rerun}{001000} \fi
314   \ifx\papermaspagespersheet\papermas@undefined \addtocounter{papermas@rerun}{010000} \fi
315   \ifx\papermasssheets\papermas@undefined \addtocounter{papermas@rerun}{100000} \fi
316 }
317

```

If any one of those commands is undefined, `papermas@rerun` is  $> 1$ , and we know that we need another compiler run.

`\AfterLastShipout` What we did not do yet, is to really *call* the command `\papermas@totmass`. We do this `\AfterLastShipout`, because we need the total number of pages, and asking for them at the end of the document might save another compilation run.

```

318 \AfterLastShipout{%
319   \papermas@totmass%
320 }%
321

```

`\AfterLastShipout` is a command from the `atveryend` package of Heiko Oberdiek, which is already loaded by my `pagesLTS` package (about how to get the `atveryend` package, please see the documentation of the `pagesLTS` package – you may need to get further packages for `pagesLTS` anyway, if they have not been installed within your  $\text{\LaTeX}$  system).

`\AtVeryEndDocument`

```

322 \AtVeryEndDocument{%
    \AtVeryEndDocument{...} is even later:

```

“The code is called after the `.aux` file is closed and read in again. It is the place for final checks, rerun hints, final messages.”

(`atveryend` package of Heiko Oberdiek, v1.5 as of 2010/03/24)

**Error code** Here it is used give a rerun warning, when it is needed:

```

323 \ifnum\value{papermas@rerun}>0
324   \PackageWarningNoLine{papermas}{!\MessageBreak%
325     Variable(s) still undefined.\MessageBreak%
326     (Error code \arabic{papermas@rerun}.)\MessageBreak%
327     Rerun to get the variable(s) right.\MessageBreak%
328   }%

```

The “Error code” can be deciphered as follows:

<code>\papermasstotal</code>	000001
<code>\papermasstotal</code>	000010
<code>\papermasformat</code>	000100
<code>\papermasmasss</code>	001000
<code>\papermaspagespersheet</code>	010000
<code>\papermasssheets</code>	100000

e.g. error code 1001 is 001001 is `\papermasmasss` and `\papermasstotal`.

If no necessity for a rerun was detected (! Check for other rerun warnings!),  
the final \PackageInfo is given:

```

329 \else
330   \message{papermas: *****}
331   \message{papermas: * This document consists of \arabic{pagesLTS.pagenr} pages. *}
332   \message{papermas: * When printing \papermaspagespersheet\space pages on one sheet of paper,
333   \message{papermas: * \papermasssheets\space sheets will be needed. *}
334   \message{papermas: * For ISO A \papermasformat\space paper of \papermasmassss\space g/m^2
335   \message{papermas: * the printout will have a mass of about \papermasstotal\space g. *}
336   \message{papermas: *****}
337   \PackageInfo{papermas}{*****\MessageBreak%
338     * This document consists of \arabic{pagesLTS.pagenr} pages. *\MessageBreak%
339     * When printing \papermaspagespersheet\space pages on one sheet of paper, *\MessageBreak%
340     * \papermasssheets\space sheets will be needed. *\MessageBreak%
341     * For ISO A \papermasformat\space paper of \papermasmassss\space g/m^2 specific mass, *\
342     * the printout will have a mass of about \papermasstotal\space g. *\MessageBreak%
343     *****\MessageBreak%
344   }%
345 \fi%
346 }
347
348 \end{package}

```

## 6 Installation

### 6.1 Downloads

Everything *should* be available on CTAN: <ftp://ftp.ctan.org/tex-archive/>, but may need additional packages themselves.

`papermas.dtx` For unpacking the `papermas.dtx` file and constructing the documentation it is required:

- T<sub>E</sub>XFormat L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, 1994/06/01, v2<sub>ε</sub>: [CTAN:](#)
- document class `ltxdoc`, 2007/11/11, v2.0u,  
[CTAN:macros/latex/base/ltxdoc.dtx](#)
- package `hltxdoc`, 2010/04/24, v0.19,  
[CTAN:macros/latex/contrib/oberdiek/hltxdoc.dtx](#)
- package `hypdoc`, 2010/03/26, v1.9,  
[CTAN:macros/latex/contrib/oberdiek/hypdoc.dtx](#)

`papermas.sty` The `papermas.sty` for L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> (i. e. all documents using the `papermas` package) requires:

- T<sub>E</sub>XFormat L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, 1994/06/01, v2<sub>ε</sub>, [CTAN:](#)
- package `kvoptions`, 2010/02/22, v3.7,  
[CTAN:macros/latex/contrib/oberdiek/kvoptions.dtx](#)
- package `pagesLTS`, 2010/06/03, v1.1b,  
[CTAN:macros/latex/contrib/muench/pagesLTS/pagesLTS.dtx](#)  
(submitted, should become available at that location soon)

`papermas-example.tex` The `papermas-example.tex` requires the same files as all documents using the `papermas` package, especially:

- package `papermas`, 2010/06/03, v1.0b,  
[CTAN:macros/latex/contrib/muench/papermas/papermas.dtx](#)  
(submitted, should become available at that location soon)  
(Well, it is the example file for this package, and because you are reading the documentation for the `papermas` package, it can be assumed that you already have some version of it – is it the current one?)

`totpages` As possible alternatives in section 3 there are listed

- package `totpages`, 2005/09/19, v2.00,  
[CTAN:macros/latex/contrib/totpages/totpages.dtx](#)

`Oberdiek` All packages of Heiko Oberdiek’s bundle ‘oberdiek’ (especially `hltxdoc`,  
`hltxdoc` `atveryend`, `kvoptions`) are also available in a TDS compliant ZIP archive:  
`atveryend` [CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#).  
`kvoptions` **Warning:** `hltxdoc`, 2010/04/24 v0.19, requires the packages

- `hypdoc`, 2010/03/26, v1.9
- `hyperref`, 2010/03/30, v6.80u (latest: 2010/05/13, v6.81e)
- `pdftexcmds`, 2010/04/01, v0.9
- `ltxcmds`, 2010/03/09, v1.4 (latest: 2010/04/26, v1.7)
- `hologo`, 2010/04/24, v1.2
- `array` (latest: 2008/09/09, v2.4c)

(or more recent versions) and does neither work with nor check for earlier versions!  
(It is probably best to download  
[CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#) and use this, because the packages in there should be both recent and compatible.)

## 6.2 Package, unpacking TDS

**Package.** This package should become available soon on [CTAN](#):

[CTAN:macros/latex/contrib/muench/papermas/papermas.dtx](#)

The source file.

[CTAN:macros/latex/contrib/muench/papermas/papermas.ins](#)

The installation file.

[CTAN:macros/latex/contrib/muench/papermas/papermas.drv](#)

The driver to generate the documentation.

[CTAN:macros/latex/contrib/muench/papermas/ltxdoc.cfg](#)

The L<sup>A</sup>T<sub>E</sub>X documentation configuration file, also for generating the documentation.

[CTAN:macros/latex/contrib/muench/papermas/papermas.pdf](#)

The documentation.

[CTAN:macros/latex/contrib/muench/papermas/papermas.sty](#)

The .sty file.

[CTAN:macros/latex/contrib/muench/papermas/papermas-example.tex](#)

The example file.

[CTAN:macros/latex/contrib/muench/papermas/papermas-example.pdf](#)

The compiled example file, as it should look like.

[CTAN:install/macros/latex/contrib/muench/papermas/papermas.tds.zip](#)

Everything in TDS compliant, compiled format.

For required other packages, see the preceding subsection.

**Unpacking.** The .dtx file is a self-extracting docstrip archive. The files are extracted by running the .dtx through plain T<sub>E</sub>X:

```
tex papermas.dtx
```

About generating the documentation see paragraph 6.4 below.

**TDS.** Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

<code>papermas.sty</code>	→ <code>tex/latex/muench/papermas.sty</code>
<code>papermas.pdf</code>	→ <code>doc/latex/muench/papermas.pdf</code>
<code>papermas-example.tex</code>	→ <code>doc/latex/muench/papermas-example.tex</code>
<code>papermas-example.pdf</code>	→ <code>doc/latex/muench/papermas-example.pdf</code>
<code>papermas.dtx</code>	→ <code>source/latex/muench/papermas.dtx</code>

If you have a `docstrip.cfg` that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

## 6.3 Refresh file name databases

If your T<sub>E</sub>X distribution (teT<sub>E</sub>X, miK<sub>T</sub>E<sub>X</sub>,...) relies on file name databases, you must refresh these. For example, teT<sub>E</sub>X users run `texhash` or `mktextlsr`.

## 6.4 Some details for the interested

**Unpacking with L<sup>A</sup>T<sub>E</sub>X.** The .dtx chooses its action depending on the format:

**plain T<sub>E</sub>X:** Run docstrip and extract the files.

**L<sup>A</sup>T<sub>E</sub>X:** Generate the documentation.

If you insist on using L<sup>A</sup>T<sub>E</sub>X for docstrip (really, docstrip does not need L<sup>A</sup>T<sub>E</sub>X), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{papermas.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

**Generating the documentation.** You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put the following line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfL<sup>A</sup>T<sub>E</sub>X:

```
pdflatex papermas.dtx
makeindex -s gind.ist papermas.idx
pdflatex papermas.dtx
makeindex -s gind.ist papermas.idx
pdflatex papermas.dtx
```

## 6.5 Compiling the example

The example file, `papermas-example.tex`, can be compiled via

```
latex papermas-example.tex
```

or (recommended)

```
pdflatex papermas-example.tex
```

but will need probably three compiler runs to get everything right.

## 7 Things suggested to be done

- Include a correct checksum for `papermas`.
- Insert the final references where to get the files at [CTAN](#).

## 8 Acknowledgements

I (H.-Martin Münch) would like to thank Heiko Oberdiek ([heiko dot oberdiek at googlemail dot com](mailto:heiko dot oberdiek at googlemail dot com)) for providing a lot (!) of usefull packages (from which I also got everything I know about creating a file in `dtx` format, ok, say it: copying), and the [news:comp.text.tex](#) and [news:de.comp.text.tex](#) newsgroups for their help in all things T<sub>E</sub>X.

## 9 History

[2010/06/01 v1.0]

- First version of this package.

[2010/06/03 v1.0b]

- New `\papermassheets` and `reruncheck` introduced; several small changes.
- Example adapted to other examples of mine.
- Updated references to other packages.
- TDS locations updated.
- Several changes in the documentation and the Readme file.

When you find a mistake or have a suggestion for an improvement of this package, please send an e-mail to the maintainer, thanks!

## 10 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; plain numbers refer to the code lines where the entry is used.

Symbols	
<code>\@auxout</code>	287, 289, 291, 293, 295, 297, 299
A	
<code>\addtocounter</code>	184, 191, 238, 256, 261, 264, 266, 310, 311, 312, 313, 314, 315
<code>\AfterLastShipout</code>	318
<code>\arabic</code>	45, 230, 233, 247, 248, 251, 255, 256, 260, 261, 270, 274, 284, 285, 326, 331, 338
<code>\AtBeginDocument</code>	307
<code>\atveryend</code>	13
<code>\AtVeryEndDocument</code>	322
D	
<code>\DeclareStringOption</code>	139, 141, 143
E	
<code>\Error_code</code>	11
F	
<code>\format</code>	3
H	
<code>\holtxdoc</code>	13
<code>\hypersetup</code>	5
I	
<code>\if@filesw</code>	286
<code>\immediate</code>	287, 289, 291, 293, 295, 297, 299
K	
<code>\kvoptions</code>	13
L	
<code>\lastpageref</code>	45
M	
<code>\markboth</code>	30
<code>\masss</code>	3
<code>\mathord</code>	152
<code>\message</code>	330, 331, 332, 333, 334, 335, 336
N	
<code>\newcommand</code>	175, 196
<code>\newcounter</code>	168, 169, 170, 171, 172, 197, 208, 220, 229, 232, 246, 250, 254, 259
O	
<code>\Oberdiek</code>	13
<code>\options</code>	3
P	
<code>\PackageError</code>	200, 212, 223
<code>\PackageInfo</code>	337
<code>\PackageWarningNoLine</code>	279, 324
<code>\pagenumbering</code>	27
<code>\pagesLTS@ifcounter</code>	275, 288
<code>\pagespersheet</code>	3
<code>\papermas-example.tex</code>	13
<code>\papermas.dtx</code>	13
<code>\papermas.sty</code>	13
<code>\papermas@format</code>	198, 205, 271
<code>\papermas@masss</code>	209, 217, 272
<code>\papermas@pagespersheet</code>	221, 226, 273
<code>\papermas@powerof</code>	175, 242
<code>\papermas@totmass</code>	196, 319
<code>\papermas@undefined</code>	308, 310, 311, 312, 313, 314, 315
<code>\papermasformat</code>	48, 159, 294, 312, 334, 341
<code>\papermasmasss</code>	48, 161, 296, 313, 334, 341
<code>\papermaspagespersheet</code>	46, 163, 298, 314, 332, 339
<code>\papermasssheets</code>	47, 165, 300, 315, 333, 340
<code>\papermasstotal</code>	50, 155, 157, 292, 310, 311, 335, 342
<code>\papermastmpformat</code>	271, 294
<code>\papermastmpmasss</code>	272, 296
<code>\papermastmppagespersheet</code>	273, 298
<code>\papermastmpr</code>	270, 292
<code>\papermastmps</code>	285, 290
<code>\papermastmpt</code>	274, 300
<code>\ProcessKeyvalOptions</code>	146
<code>\ProvidesPackage</code>	117
R	
<code>\RequirePackage</code>	131, 132
<code>\rm</code>	152
S	
<code>\section</code>	29
<code>\setcounter</code>	173, 176, 177, 179, 180, 182, 188, 198, 209, 221, 230, 233, 247, 248, 251, 255, 260, 267, 284, 290, 309
<code>\SetupKeyvalOptions</code>	138
T	
<code>\totpages</code>	13
U	
<code>\unit</code>	35, 48, 50, 54, 55, 148, 152
W	
<code>\write</code>	287, 289, 291, 293, 295, 297, 299