

TRANSLATIONS

vo.9a 2013/04/08

part of the **ExSHEETS** bundle

a simple translator

Clemens NIEDERBERGER

<https://bitbucket.org/cgnieder/exsheets/>
contact@mychemistry.eu

English documentation

Contents

	3.2	A Small Example	3
	3.3	Usage in Packages	4
	3.4	Dictionaries	4
1	Motivation	1	
2	License and Requirements	1	
3	Usage	2	
3.1	Available Commands	2	
	4	Implementation	5
		Index	13

1 Motivation

I wrote this package when I was in need for an expandable version of translator's `\translate`. If you don't need that then there is very likely no need whatsoever for this package as translator from the beamer¹ bundle provides very similar functionality **TRANSLATIONS** does (except for: translator's `\translate` isn't expandable, `\GetTranslation` is.).

2 License and Requirements

TRANSLATIONS is placed under the terms of the L^AT_EX Project Public License, version 1.3 or later (<http://www.latex-project.org/lppl.txt>). It has the status "maintained."

TRANSLATIONS requires the etoolbox² package.

¹ CTAN: beamer ² CTAN: etoolbox

3 Usage

3.1 Available Commands

Below the commands provided by **TRANSLATIONS** are explained. The symbol \triangleright means that the command is expandable, \blacktriangleright means that it isn't.

- \blacktriangleright **\DeclareLanguage**{<lang>}
 Declare a language that can be used by **TRANSLATIONS**. If the language already exists it will be silently redefined. This command can only be used in the preamble.
- \blacktriangleright **\DeclareLanguageAlias**{<lang2>}{<lang1>}
 Declares <lang2> to be an alias of <lang1>. If <lang1> doesn't exist yet a warning will be raised and it will be defined. This command can only be used in the preamble.
- \blacktriangleright **\DeclareLanguageDialect**{<dialect>}{<lang>}
 Declares <dialect> to be a dialect of language <lang>. If a translation for <dialect> is provided it is used by the translation macros. If there is none the corresponding translation for <lang> is used instead.
- \blacktriangleright **\NewTranslation**{<lang>}{<key>}{<translation>}
 Defines a translation of key <key> for the language <lang>. An error will be raised if a translation of <key> already exists. This command can only be used in the preamble.
- \blacktriangleright **\RenewTranslation**{<lang>}{<key>}{<translation>}
 Redefines a translation of key <key> for the language <lang>. An error will be raised if no translation of <key> exists. This command can only be used in the preamble.
- \blacktriangleright **\DeclareTranslation**{<lang>}{<key>}{<translation>}
 Defines a translation of key <key> for the language <lang>. No error will be raised if a translation of <key> already exists. This command can only be used in the preamble.
- \blacktriangleright **\DeclareTranslationFallback**{<key>}{<fallback>}
 Defines a fallback translation for key <key> that is used in case no translation of <key> for the currently active language has been provided. No error will be raised if a fallback for <key> already exists. This command can only be used in the preamble.
- \triangleright **\GetTranslationFor**{<lang>}{<key>}
 Fetches and prints the translation of <key> for the language <lang>. This command is expandable.
- \triangleright **\GetTranslation**{<key>}
 Fetches and prints the translation of <key> for the currently active language (as for example set by babel³). This command is expandable.

³ CTAN: babel

- `\SaveTranslationFor{<cmd>}{<lang>}{<key>}`
Fetches and saves the translation of <key> for the language <lang> in the macro <cmd>.
- `\SaveTranslation{<cmd>}{<key>}`
Fetches and saves the translation of <key> for the currently active language (as for example set by babel) in the macro <cmd>.
- `\LoadDictionary{<name>}`
Loads a file named <name>-<lang>.trsl where <lang> corresponds to the lowercase name of the current language as defined with `\DeclareLanguage`. This file should contain the translations for the specified language.
- `\LoadDictionaryFor{<lang>}{<name>}`
Loads a file named <name>-<lang>.trsl.
- `\DeclareDictTranslation{<key>}{<translation>}`
This command is to be used in a dictionary file and picks up the language of that file, see section 3.4 for an example.

Quite a number of languages already are defined, either directly or via an alias. So, before you define a language you should take a look in `translations.sty` if the language doesn't already exist.

3.2 A Small Example

Here is a small example of usage:

```

1 % in the preamble:
2 % \DeclareTranslation{English}{Kueche}{kitchen}
3 % \DeclareTranslation{German}{Kueche}{K\"uche}
4 % \DeclareTranslation{Spanish}{Kueche}{cocina}
5 % \DeclareTranslation{French}{Kueche}{cuisine}
6
7 \GetTranslation{Kueche}
8 \SaveTranslation\kitchen{Kueche}
9 \SaveTranslationFor\cuisine{french}{Kueche}
10
11 \selectlanguage{ngerman}
12 \GetTranslation{Kueche} \kitchen\ \GetTranslationFor{spanish}{Kueche}
13 \cuisine

kitchen
Küche kitchen cocina cuisine

```

There is also support for dialects:

```

1 % in the preamble:
2 % \DeclareTranslation{English}{foo}{foo}
3 % \DeclareTranslation{British}{foo}{bar}
4 \GetTranslationFor{English}{foo} \\
5 \GetTranslationFor{British}{foo} \\
6 \GetTranslationFor{American}{foo}

foo
bar
foo

```

3.3 Usage in Packages

In package a typical usage in a package would look as follows:

```

1 \RequirePackage{translations}
2 \DeclareTranslationFallback{mypackage-title}{Nice Title}
3 \DeclareTranslation{English}{mypackage-title}{Nice Title}
4 \DeclareTranslation{French}{mypackage-title}{Beau Titre}
5 \DeclareTranslation{German}{mypackage-title}{Sch\{"o}ner
  Titel}
6 ...
7 \def\mypackage@title{\GetTranslation{mypackage-title}}

```

3.4 Dictionaries

A typical dictionary file should look as follows:

```

1 % this is file housing-german.trsl
2 \ProvideDictionaryFor{German}{housing}[<version info>]
3 \DeclareDictTranslation{kitchen (housing)}{K\uche}
4 \DeclareDictTranslation{bathroom (housing)}{Bad}
5 \DeclareDictTranslation{living room (housing)}{Wohnzimmer}
6 \DeclareDictTranslation{bedroom (housing)}{Schlafzimmer}
7 ...
8 \endinput

```

4 Implementation

In the following code the lines 1–30 have been omitted. They only repeat the license statement which has already been mentioned in section 2.

```

31 \def\@trnslt@date{2013/04/08}
32 \def\@trnslt@version{v0.9a}
33
34 \ProvidesPackage{translations}[\@trnslt@date\space \@trnslt@version\space a
    simple translator]
35 \RequirePackage{etoolbox}
36
37 % -----
38
38 % message handling
39 \def\@trnslt@error@message{%
40     For details have a look at the 'exsheets' manual.}
41
42 \def\@trnslt@create@message#1{%
43     \ifstrequal{#1}{Error}
44     {%
45         \lowercase{\csdef{\@trnslt@#1}}##1{%
46             \csuse{Package#1}{translations}{##1}{\@trnslt@error@message}}%
47         }{%
48             \lowercase{\csdef{\@trnslt@#1}}##1{%
49                 \csuse{Package#1}{translations}{##1}}%
50         }}
51 \@trnslt@create@message{Error}
52 \@trnslt@create@message{Warning}
53 \@trnslt@create@message{WarningNoLine}
54 \@trnslt@create@message{Info}
55
56 \def\@trnslt@err@unknown@lang#1{%
57     \@trnslt@error{Unknown language '#1'}}
58
59 \def\@trnslt@warn@unknown@lang#1{%
60     \@trnslt@warning{Unknown language '#1'}}
61
62 \def\@trnslt@err@already@defined#1#2{%
63     \@trnslt@error{The #2 translation for '#1' is already defined.}}
64
65 \def\@trnslt@err@not@defined#1#2{%
66     \@trnslt@error{The \@trnslt@language{#2} translation for '#1' is not
        defined yet.}}
67
68 % -----

```

4 Implementation

```

69 % check if babel or polyglossia is used
70 \AtEndPreamble{
71   \@ifpackageloaded{babel}{}{
72     \@ifpackageloaded{polyglossia}{}
73     {\@trnslt@warning{No language package found. I am going to use '
       english'
       as default language.}}
74   }
75   \ifdef\languagename{}
76     {\def\languagename{english}}
77   \def\@trnslt@current@language{\languagename}
78   \ifdef\bbl@afterfi{}
79     {\long\def\bbl@afterfi#1\fi{\fi#1}}
80 }
81
82
83 % -----
84
84 % \DeclareLanguage and \DeclareLanguageAlias
85 % #1: language
86 \newrobustcmd*\DeclareLanguage[1]{%
87   \@trnslt@declare@language{#1}}
88 \@onlypreamble\DeclareLanguage
89
90 \def\@trnslt@declare@language#1{%
91   \csdef{@trnslt@language@#1}{#1}}
92
93 % #1: alias
94 % #2: language
95 \newrobustcmd*\DeclareLanguageAlias[2]{%
96   \@trnslt@declare@languagealias{#1}{#2}}
97 \@onlypreamble\DeclareLanguageAlias
98
99 \def\@trnslt@declare@languagealias#1#2{%
100   \ifcsdef{@trnslt@language@#2}
101     {}{%
102       \@trnslt@warn@unknown@lang{#2}%
103       \@trnslt@declare@language{#2}%
104     }%
105   \lowercase{\@trnslt@if@dialect{#2}}{#2}
106   {%
107     \csletcs
108       {@trnslt@language@\@trnslt@language{#2}@dialect@#1}
109       {@trnslt@language@\@trnslt@language{#2}@dialect@#2}%
110   }
111   {}%
112   \csletcs{@trnslt@language@#1}{@trnslt@language@#2}}
113

```

4 Implementation

```

114 % #1: dialect
115 % #2: language
116 \newrobustcmd*\DeclareLanguageDialect[2]{%
117   \@trnslt@declare@languagedialect{#1}{#2}}
118 \@onlypreamble\DeclareLanguageDialect
119
120 \def\@trnslt@declare@languagedialect#1#2{%
121   \ifcsdef{\@trnslt@language@#2}
122     {}{%
123       \@trnslt@warn@unknown@lang{#2}%
124       \@trnslt@declare@language{#2}%
125     }%
126   \csdef{\@trnslt@language@\@trnslt@language{#2}@dialect@#1}{}%
127   \csletcs{\@trnslt@language@#1}{\@trnslt@language@#2}}
128
129 \def\@trnslt@if@dialect#1#2{%
130   \ifcsdef{\@trnslt@language@\@trnslt@language{#2}@dialect@#1}
131     {\expandafter\@firstoftwo}
132     {\expandafter\@secondoftwo}}
133
134 \def\@trnslt@language#1{%
135   \csuse{\@trnslt@language@#1}}
136
137 % dummy language: 'fallback':
138 \DeclareLanguage{fallback}
139 \DeclareLanguageAlias{Fallback}{fallback}
140
141 % -----
142
142 % \DeclareTranslation, \NewTranslation and \RenewTranslation
143 % #1: language
144 % #2: word
145 % #3: replacement
146 \newrobustcmd*\DeclareTranslation[3]{%
147   \@trnslt@declare@translation{#2}{#1}{#3}}
148 \@onlypreamble\DeclareTranslation
149
150 \newrobustcmd*\DeclareTranslationFallback[2]{%
151   \@trnslt@declare@translation{#1}{fallback}{#2}}
152 \@onlypreamble\DeclareTranslationFallback
153
154 \def\@trnslt@declare@translation#1#2#3{%
155   \ifcsdef{\@trnslt@language@#2}
156     {%
157       \csdef{\@trnslt@word@#1@literal}{#1}%
158       \lowercase{\@trnslt@if@dialect{#2}}{#2}
159       {\csdef{\@trnslt@word@#1@#2}{#3}}

```

4 Implementation

```

160         {}%
161         \ifcsdef{@trnslt@word@#1@\@trnslt@language{#2}}
162         {}
163         {\csdef{@trnslt@word@#1@\@trnslt@language{#2}}{#3}}%
164     }
165     {\@trnslt@err@unknown@lang{#2}}}}
166
167 \def\@trnslt@if@translation#1#2{%
168     \ifcsdef{@trnslt@word@#1@\@trnslt@language{#2}}
169     {\expandafter\@firstoftwo}
170     {\expandafter\@secondoftwo}}
171
172 \newrobustcmd*\NewTranslation[3]{%
173     \@trnslt@new@translation{#2}{#1}{#3}}
174 \onlypreamble\NewTranslation
175
176 \def\@trnslt@new@translation#1#2#3{%
177     \@trnslt@if@translation{#1}{#2}
178     {\@trnslt@err@already@defined{#1}{#2}}
179     {\@trnslt@declare@translation{#1}{#2}{#3}}}
180
181 \newrobustcmd*\RenewTranslation[3]{%
182     \@trnslt@renew@translation{#2}{#1}{#3}}
183 \onlypreamble\RenewTranslation
184
185 \def\@trnslt@renew@translation#1#2#3{%
186     \@trnslt@if@translation{#1}{#2}
187     {\@trnslt@declare@translation{#1}{#2}{#3}}
188     {\@trnslt@err@not@defined{#1}{#2}}}
189
190 % -----
191
192 % \GetTranslationFor and \GetTranslation
193 % these need to be expandable!
194 % #1: language
195 % #2: word
196 \newcommand*\GetTranslationFor[2]{%
197     \@trnslt@get@translation@for{#2}{#1}}
198
199 \newcommand*\GetTranslation[1]{%
200     \@trnslt@get@translation@for{#1}{\@trnslt@current@language}}
201
202 % #1: word #2: language
203 \def\@trnslt@get@translation@for#1#2{%
204     \@trnslt@if@translation{#1}{#2}
205     {%
206         \@trnslt@if@dialect{#2}{#2}

```


4 Implementation

```

206     {%
207         \ifcsdef{@trnslt@word@#1@#2}
208         {\csuse{@trnslt@word@#1@#2}}
209         {\csuse{@trnslt@word@#1@\@trnslt@language{#2}}}%
210     }
211     {\csuse{@trnslt@word@#1@\@trnslt@language{#2}}}%
212 }
213 {%
214     \@trnslt@warning{Translation for ‘#1’ in #2 unknown. You may try to
use
215         \string\DeclareTranslation{#2}{#1}{ ... } in your preamble.}%
216     \@trnslt@if@translation{#1}{fallback}
217     {%
218         \@trnslt@info{Using fallback translation for ‘#1’}%
219         \csuse{@trnslt@word@#1@fallback}
220     }%
221     {\csuse{@trnslt@word@#1@literal}}%
222 }}
223
224 % -----
225
226 % \SaveTranslationFor and \SaveTranslation
227 \newrobustcmd*\SaveTranslationFor[3]{%
228     \@trnslt@save@translation@for{#1}{#3}{#2}}
229
230 \newrobustcmd*\SaveTranslation[2]{%
231     \@trnslt@save@translation@for{#1}{#2}{\@trnslt@current@language}}
232
233 \def\@trnslt@save@translation@for#1#2#3{%
234     \edef#1{%
235         \@trnslt@if@translation{#2}{#3}
236         {\csuse{@trnslt@word@#2@\@trnslt@language{#3}}}
237         {}%
238     }%
239 }
240
241 % -----
242
243 % \LoadDictionary and \LoadDictionaryFor
244 \newrobustcmd*\LoadDictionary[1]{%
245     \@trnslt@load@dictionary@for{#1}{\@trnslt@current@language}}
246 \@onlypreamble\LoadDictionary
247
248 \newrobustcmd*\LoadDictionaryFor[2]{%
249     \@trnslt@load@dictionary@for{#2}{#1}}
250 \@onlypreamble\LoadDictionaryFor
251
252 % #1: name

```

4 Implementation

```

250 % #2: lang
251 \def\@trnslt@load@dictionary@for#1#2{%
252   \AtBeginDocument{%
253     \InputIfFileExists{#1-\@trnslt@language{#2}.trsl}
254     {\@trnslt@info{loading dictionary '#1' for '#2'.}}
255     {\@trnslt@warning{File '#1-\@trnslt@language{#2}.trsl' not found.}}%
256   }}
257
258 \newrobustcmd*\ProvideDictionaryFor[2]{%
259   \@trnslt@provide@dictionary@for{#1}{#2}}
260 \onlypreamble\ProvideDictionaryFor
261
262 \def\@trnslt@provide@dictionary@for#1#2{%
263   \def\@trnslt@dictionary@name{#2}%
264   \def\@trnslt@dictionary@lang{#1}%
265   \@ifnextchar[
266     {\@trnslt@provide@dictionary@version}
267     {\ProvidesFile{#2-#1.trsl}[{#1 translation file '#2'}]}}
268
269 \def\@trnslt@provide@dictionary@version[#1]{%
270   \ProvidesFile
271     {\@trnslt@dictionary@name-\@trnslt@dictionary@lang.trsl}%
272     [(\@trnslt@dictionary@lang\space translation file '\@
273       trnslt@dictionary@name') #1]}
274
275 % \@trnslt@dictionary@language
276 \newrobustcmd*\DeclareDictTranslation[2]{%
277   \@trnslt@declare@translation{#1}{\@trnslt@dictionary@lang}{#2}}
278 \onlypreamble\DeclareDictTranslation
279
280 % predefined languages
281 \DeclareLanguage{albanian}
282 \DeclareLanguage{bulgarian}
283 \DeclareLanguage{catalan}
284 \DeclareLanguage{croatian}
285 \DeclareLanguage{czech}
286 \DeclareLanguage{danish}
287 \DeclareLanguage{dutch}
288 \DeclareLanguage{english}
289 \DeclareLanguage{finnish}
290 \DeclareLanguage{french}
291 \DeclareLanguage{german}
292 \DeclareLanguage{greek}
293 \DeclareLanguage{hebrew}
294 \DeclareLanguage{hungarian}

```

4 Implementation

```
295 \DeclareLanguage{icelandic}
296 \DeclareLanguage{italian}
297 \DeclareLanguage{norwegian}
298 \DeclareLanguage{polish}
299 \DeclareLanguage{portuges}
300 \DeclareLanguage{romanian}
301 \DeclareLanguage{russian}
302 \DeclareLanguage{serbocroatian}
303 \DeclareLanguage{slovak}
304 \DeclareLanguage{slovenian}
305 \DeclareLanguage{spanish}
306 \DeclareLanguage{swedish}
307 \DeclareLanguage{turkish}
308 \DeclareLanguage{ukrainian}
309
310 \DeclareLanguageAlias {Albanian}{albanian}
311 \DeclareLanguageAlias {Bulgarian}{bulgarian}
312 \DeclareLanguageAlias {Catalan}{catalan}
313 \DeclareLanguageAlias {Croatian}{croatian}
314 \DeclareLanguageAlias {Czech}{czech}
315 \DeclareLanguageAlias {Danish}{danish}
316 \DeclareLanguageAlias {Dutch}{dutch}
317 \DeclareLanguageAlias {Finnish}{finnish}
318 \DeclareLanguageAlias {français}{french}
319 \DeclareLanguageAlias {Français}{français}
320 \DeclareLanguageDialect{canadien}{french}
321 \DeclareLanguageAlias {Canadien}{canadien}
322 \DeclareLanguageAlias {French}{french}
323 \DeclareLanguageDialect{american}{english}
324 \DeclareLanguageAlias {American}{american}
325 \DeclareLanguageDialect{australian}{english}
326 \DeclareLanguageAlias {Australian}{australian}
327 \DeclareLanguageDialect{british}{english}
328 \DeclareLanguageAlias {British}{british}
329 \DeclareLanguageDialect{canadian}{english}
330 \DeclareLanguageAlias {Canadian}{canadian}
331 \DeclareLanguageAlias {English}{english}
332 \DeclareLanguageDialect{UKenglish}{english}
333 \DeclareLanguageDialect{USenglish}{english}
334 \DeclareLanguageDialect{austrian}{german}
335 \DeclareLanguageAlias {Austrian}{austrian}
336 \DeclareLanguageAlias {German}{german}
337 \DeclareLanguageAlias {germanb}{german}
338 \DeclareLanguageDialect{naustrian}{austrian}
339 \DeclareLanguageAlias {ngerman}{german}
340 \DeclareLanguageAlias {Greek}{greek}
341 \DeclareLanguageAlias {polutonikogreek}{greek}
```

4 Implementation

```
342 \DeclareLanguageAlias {Hebrew}{hebrew}
343 \DeclareLanguageAlias {Hungarian}{hungarian}
344 \DeclareLanguageDialect{magyar}{hungarian}
345 \DeclareLanguageAlias {Magyar}{magyar}
346 \DeclareLanguageAlias {Icelandic}{icelandic}
347 \DeclareLanguageAlias {Italian}{italian}
348 \DeclareLanguageAlias {norsk}{norwegian}
349 \DeclareLanguageAlias {Norsk}{norsk}
350 \DeclareLanguageAlias {Norwegian}{norwegian}
351 \DeclareLanguageAlias {nynorsk}{norwegian}
352 \DeclareLanguageAlias {Nynorsk}{nynorsk}
353 \DeclareLanguageAlias {Polish}{polish}
354 \DeclareLanguageDialect{brazil}{portuges}
355 \DeclareLanguageAlias {Brazil}{brazil}
356 \DeclareLanguageAlias {brazilian}{brazil}
357 \DeclareLanguageAlias {Brazilian}{brazilian}
358 \DeclareLanguageAlias {Portuges}{portuges}
359 \DeclareLanguageAlias {portuguese}{portuges}
360 \DeclareLanguageAlias {Portuguese}{portuguese}
361 \DeclareLanguageAlias {Romanian}{romanian}
362 \DeclareLanguageAlias {Russian}{russian}
363 \DeclareLanguageAlias {Serbocroatian}{serbocroatian}
364 \DeclareLanguageAlias {Slovak}{slovak}
365 \DeclareLanguageAlias {Slovenian}{slovenian}
366 \DeclareLanguageAlias {Spanish}{spanish}
367 \DeclareLanguageAlias {Swedish}{swedish}
368 \DeclareLanguageAlias {Turkish}{turkish}
369 \DeclareLanguageAlias {Ukrainian}{ukrainian}
370
371 \endinput
372
373 % HISTORY:
374 2012/09/30 v0.2beta - first version (as part of the 'exsheets' bundle)
375 2012/10/05 v0.2 - \LoadDictionary and \LoadDictionaryFor added and
376 loads of languages defined.
377 2013/03/10 v0.8 - basic dictionaries for English, German, French and
378 Spanish
379 2013/04/04 v0.8a - new command \DeclareDictTranslation
380 2013/04/07 v0.9 - bug fix in \DeclareDictTranslation
381 2013/04/08 v0.9a - slightly improved messages
382 - changed fallback warning into info
383 - synchronized version number with 'exsheets' until now
but
won't any more
```

Index

Section titles are indicated **bold**, packages sans serif, commands \brown and options yellow.

B	G	R
babel.....2 f.	\GetTranslation.....1 f.	\RenewTranslation.....2
beamer.....1	\GetTranslationFor.....2	Requirements1
D	L	S
\DeclareDictTranslation 3	\LoadDictionary.....3	\SaveTranslation.....3
\DeclareLanguage.....2 f.	\LoadDictionaryFor.....3	\SaveTranslationFor.....3
\DeclareLanguageAlias .. 2		
\DeclareLanguageDialect 2		
\DeclareTranslation.....2	M	T
\DeclareTranslationFallback	Motivation1	\translate.....1
2		translator.....1
E	N	U
etoolbox.....1	\NewTranslation.....2	Usage2 ff.