

tracklang v1.0: tracking language options

Nicola L. C. Talbot

<http://www.dickimaw-books.com/>

2014-09-29

Abstract

The tracklang package is provided for package developers who want a simple interface to find out which languages the user has requested through packages such as babel and polyglossia. *This package doesn't provide any translations.* Its purpose is simply to track which languages have been requested by the user. Generic T_EX code is in `tracklang.tex` for non- \LaTeX users (category code of @ must be 11 on input).

Contents

1	Introduction	1
2	Detecting the User's Requested Languages	2
2.1	Examples	10
2.1.1	animals.sty	10
2.1.2	regions.sty	20
3	Adding Support for Language Tracking	25
4	Plain T_EX Example	26
5	The Code	28
5.1	\LaTeX Code (<code>tracklang.sty</code>)	28
5.2	Generic Code (<code>tracklang.tex</code>)	33
	Index	66

1 Introduction

When I'm developing a package that provides multilingual support (for example, glossaries) it's cumbersome trying to work out if the user has requested

translations for fixed text. This usually involves checking if `babel` or `ngerman` or `translator` or `polyglossia` has been loaded and, if so, what language settings have been used. The result can be a tangled mass of conditional code. The alternative is to tell users to add the language as a document class option, which they may or may not want to do, or to tell them to supply the language settings to every package they load that provides multilingual support, which users are even less likely to want to do.

The `tracklang` package tries to neaten this up by working out as much of this information as possible for you and providing a command that iterates through the loaded languages. This way, you can just iterate through the list of tracked languages and, for each language, either define the translations or warn the user that there's no translation for that language.

This package works best with `ngerman` and recent versions of `babel`, and next best when the language options are specified in the document class option list. It works fairly well with `translator` but will additionally assume the root language was also requested when a dialect is specified. So, for example,

```
\usepackage[british]{translator}  
\usepackage{tracklang}
```

is equivalent to

```
\usepackage[british]{translator}  
\usepackage[english,british]{tracklang}
```

This means that `\ForEachTrackedDialect` will iterate through the list “`english,british`” instead of just “`british`”.

Unfortunately I can't work out how to pick up the language variant from `polyglossia`, so only the root languages are detected, which is suboptimal but at least provides some information.

If the `ngerman` package has been loaded, `tracklang` sets the root language to “`german`” and the dialect to “`ngerman`”.

Section 2 is for package writers who want to add multilingual support to their package and need to know which settings the user has requested through language packages like `babel`. Section 3 is for developers of language definition packages who want to help other package writers to detect what languages have been requested. Section 4 provides a plain `TEX` example.

2 Detecting the User's Requested Languages

The `tracklang` package tries to track the loaded languages and the option names used to identify those languages. For want of a better term, the language option names are referred to as dialects even if they're only a synonym for the language rather than an actual dialect. For example, if the user has requested `british`, the root language label is `english` and the dialect is `british`, whereas if the

user requested UKenglish, the root language label is english and the dialect is UKenglish.

In addition to the root language label and the dialect identifier, most of the language options also have corresponding ISO codes. In most cases there is an ISO 639-1 or an ISO 639-2 code (or both), and in some cases there is an ISO 3166-1 code identifying the dialect region.

When tracklang is loaded, it first attempts to find the language options through the package options supplied to tracklang. This means that any languages that have been supplied in the document class options should get identified. If no languages have been supplied in this way, tracklang then attempts to identify any babel language options and failing that it will try the translator language options. It will then check if ngerman or polyglossia have been loaded.

Each identified language and dialect is added to the *tracked language* and *tracked dialect* lists. Note that the tracked language and tracked dialect are labels rather than proper nouns. If a dialect label is identical to its root language label, the label will appear in both lists.

You can check whether or not any languages have been detected using:

```
\AnyTrackedLanguages{<true part>}{<false part>}
```

This will do <true part> if one or more languages have been detected otherwise it will do <false part>. (Each detected dialect will automatically have the root language label added to the tracked language list, if it's not already present.)

You can iterate through each tracked dialect using:

```
\ForEachTrackedDialect{<cs>}{<code>}
```

At the start of each iteration, this sets <cs> to the tracked dialect and does <code>.

You can iterate through each tracked language using:

```
\ForEachTrackedLanguage{<cs>}{<code>}
```

At the start of each iteration, this sets <cs> to the tracked language and does <code>.

You can test if a root language has been detected using:

```
\IfTrackedLanguage{<label>}{<true part>}{<false part>}
```

where <label> is the language label. If true, this does <true part> otherwise it does <false part>.

You can test if a particular dialect has been detected using:

```
\IfTrackedDialect{<label>}{<true part>}{<false part>}
```

where *<label>* is the dialect label. If the root language was explicitly specified, then it will also be detected as a dialect.

For example:

```
\documentclass[british,dutch]{article}

\usepackage{tracklang}

\begin{document}
‘‘english’’ \IfTrackedDialect{english}{has}{hasn’t} been specified.

‘‘british’’ \IfTrackedDialect{british}{has}{hasn’t} been specified.

‘‘flemish’’ \IfTrackedDialect{flemish}{has}{hasn’t} been specified.

‘‘dutch’’ \IfTrackedDialect{dutch}{has}{hasn’t} been specified.

‘‘english’’ or an English variant
\IfTrackedLanguage{english}{has}{hasn’t} been specified.

\end{document}
```

This produces:

“english” hasn’t been specified.
“british” has been specified.
“flemish” hasn’t been specified.
“dutch” has been specified.
“english” or an English variant has been specified.

You can find the root language label for a given tracked dialect using:

```
\TrackedLanguageFromDialect{<dialect>}
```

If *<dialect>* hasn’t been defined this does nothing otherwise it expands to the root language label.

You can find the tracked dialects from a given root language using:

```
\TrackedDialectsFromLanguage{<root language label>}
```

This will expand to a comma-separated list of dialect labels if the root language label has been defined, otherwise it does nothing.

You can test if a language (or dialect) has a corresponding ISO code using:

```
\IfTrackedLanguageHasIsoCode{<code type>}{<label>}{<true part>}{<false part>}
```

where *<code type>* is the type of ISO code (for example, 639-1 for root languages or 3166-1 for regional dialects), and *<label>* is the language or dialect label.

Alternatively, you can test if a particular ISO code has been defined using:

```
\IfTrackedIsoCode{<code type>}{<code>}{<true part>}{<false part>}
```

where *<code type>* is again the type of ISO code (for example, 639-1 or 3166-1), and *<code>* is the particular code (for example, en for ISO 639-1 or GB for ISO 3166-1).

You can fetch the language (or dialect) label associated with a given ISO code using:

```
\TrackedLanguageFromIsoCode{<code type>}{<code>}
```

This does nothing if the given *<code>* for the given ISO *<code type>* has not been defined, otherwise it expands a comma-separated list of language or dialect labels.

You can fetch the ISO code for a given code type using:

```
\TrackedIsoCodeFromLanguage{<code type>}{<label>}
```

where *<label>* is the language or dialect label and *<code type>* is the ISO code type (for example, 639-1 or 3166-1). Unlike `\TrackedLanguageFromIsoCode`, this command only expands to a single label rather than a comma-separated list.

The above commands do nothing in the event of an unknown code or code type, so if you accidentally get the wrong code type, you won't get an error. If you're unsure of the code type, you can use the following commands:

```
\TwoLetterIsoCountryCode
```

This expands to 3166-1 and is used for the two-letter country codes.

```
\TwoLetterIsoLanguageCode
```

This expands to 639-1 and is used for the two-letter root language codes.

```
\ThreeLetterIsoLanguageCode
```

This expands to 639-2 and is used for the three-letter root language codes.

For example:

```
\documentclass[british,français,american,canadian,canadien,dutch]{article}
```

```
\usepackage{tracklang}
```

```

\begin{document}

Languages: \ForEachTrackedLanguage{\ThisLanguage}{\ThisLanguage\space
(ISO \TwoLetterIsoLanguageCode:
‘‘\TrackedIsoCodeFromLanguage{\TwoLetterIsoLanguageCode}{\ThisLanguage}’’). }

Dialects: \ForEachTrackedDialect{\ThisDialect}{\ThisDialect\space
(\IfTrackedLanguageHasIsoCode{\TwoLetterIsoCountryCode}{\ThisDialect}%
{ISO \TwoLetterIsoCountryCode:
‘‘\TrackedIsoCodeFromLanguage{\TwoLetterIsoCountryCode}{\ThisDialect}’’}%
{no specific region};
root: \TrackedLanguageFromDialect{\ThisDialect}). }

Language for ISO \TwoLetterIsoCountryCode\ ‘‘GB’’:
\TrackedLanguageFromIsoCode{\TwoLetterIsoCountryCode}{GB}.

Language for ISO \TwoLetterIsoCountryCode\ ‘‘CA’’:
\TrackedLanguageFromIsoCode{\TwoLetterIsoCountryCode}{CA}.

Country ISO \TwoLetterIsoCountryCode\ code for ‘‘canadian’’:
\TrackedIsoCodeFromLanguage{\TwoLetterIsoCountryCode}{canadian}.

\end{document}

```

This produces:

Languages: english (ISO 639-1: “en”). french (ISO 639-1: “fr”).
dutch (ISO 639-1: “nl”).

Dialects: american (ISO 3166-1: “US”; root: english). british (ISO
3166-1: “GB”; root: english). canadian (ISO 3166-1: “CA”; root: en-
glish). canadien (ISO 3166-1: “CA”; root: french). dutch (no specific
region; root: dutch). francais (no specific region; root: french).

Language for ISO 3166-1 “GB”: british.

Language for ISO 3166-1 “CA”: canadian,canadien.

Country ISO 3166-1 code for “canadian”: CA.

Most packages that implement multilingual support have a set of language definition files for each supported language or dialect. It may be that only the root language is needed, if there are no variations between that language’s dialect (for the purposes of that package), or it may be that separate definition files are required for each dialect. However it can be awkward trying to map the requested dialect or language label to the file name. Should, say, the file containing the French code be called $\langle prefix \rangle$ -french- $\langle suffix \rangle$ or $\langle prefix \rangle$ -frenchb- $\langle suffix \rangle$ or $\langle prefix \rangle$ -francais- $\langle suffix \rangle$? Should, say, the file containing the British English code be called $\langle prefix \rangle$ -british- $\langle suffix \rangle$ or $\langle prefix \rangle$ -UKenglish- $\langle suffix \rangle$? If you want to modularise the language support

for your package so that each language module has a different maintainer will the maintainers know what tag to use for their language?

To help with this, tracklang provides:

```
\IfTrackedLanguageFileExists{<label>}<prefix>{<suffix>}{<true part>}{<false part>}
```

This attempts to find the file called $\langle prefix \rangle \langle tag \rangle \langle suffix \rangle$ where $\langle tag \rangle$ is determined from $\langle label \rangle$. If the file is found

```
\CurrentTrackedTag
```

is set to $\langle tag \rangle$ and $\langle true part \rangle$ is done, otherwise $\langle false part \rangle$ is done.

The $\langle tag \rangle$ is determined as follows:

1. If no dialect with the given label has been detected, the condition evaluates to *false* and `\CurrentTrackedTag` is empty.
2. If $\langle label \rangle$ has an ISO 3166-1 code, then

```
\CurrentTrackedRegion
```

is set to that code otherwise it's empty.

3. If the file $\langle prefix \rangle \langle label \rangle \langle suffix \rangle$ exists, $\langle tag \rangle$ (`\CurrentTrackedTag`) is just $\langle label \rangle$ and the condition evaluates to *true*.
4. The root language label is obtained and stored in

```
\CurrentTrackedLanguage
```

(This may or may not be the same as the original dialect $\langle label \rangle$.)

5. If the root language has an ISO 639-1 code, this code is fetched and stored in

```
\CurrentTrackedIsoCode
```

If the root language doesn't have an ISO 639-1 code, but does have an ISO 639-2 code, then this code is fetched and stored in `\CurrentTrackedIsoCode`.

6. If is no ISO 639-1 and no ISO 636-2 code:
 - a) If there is also no ISO 3166-1 code (`\CurrentTrackedRegion` is empty), then $\langle tag \rangle$ (`\CurrentTrackedTag`) is set to the root language (`\CurrentTrackedLanguage`). If the file $\langle prefix \rangle \langle tag \rangle \langle suffix \rangle$

exists, the condition will evaluate to *true* otherwise it will evaluate to *false*.

- b) If there is an ISO 3166-1 code, then $\langle tag \rangle$ ($\backslash\text{CurrentTrackedTag}$) will be set to $\backslash\text{CurrentTrackedRegion}$ and if the file $\langle prefix \rangle \langle tag \rangle \langle suffix \rangle$ exists, the condition will evaluate to *true* otherwise it will evaluate to *false*.

- 7. If $\backslash\text{CurrentTrackedRegion}$ is empty (no ISO 3166-1 code) then $\langle tag \rangle$ ($\backslash\text{CurrentTrackedTag}$) is set to just $\backslash\text{CurrentTrackedIsoCode}$ and if the file $\langle prefix \rangle \{ \langle tag \rangle \} \langle suffix \rangle$ exists, the condition will evaluate to *true*. If the file doesn't exist and $\backslash\text{CurrentTrackedIsoCode}$ was set to the ISO 639-1 code, then the test will be repeated with the ISO 639-2 code. If that test also fails, then the condition evaluates to *false*.

- 8. The $\langle tag \rangle$ ($\backslash\text{CurrentTrackedTag}$) is then set to

$\backslash\text{CurrentTrackedIsoCode} - \backslash\text{CurrentTrackedRegion}$

and if the file $\langle prefix \rangle \langle tag \rangle \langle suffix \rangle$ exists, the condition evaluates to *true*.

- 9. The $\langle tag \rangle$ ($\backslash\text{CurrentTrackedTag}$) is then set to just $\backslash\text{CurrentTrackedIsoCode}$, and if the file $\langle prefix \rangle \langle tag \rangle \langle suffix \rangle$ exists, the condition evaluates to *true*. If the file doesn't exist and $\backslash\text{CurrentTrackedIsoCode}$ is set to the ISO 636-1 code and the language also has an ISO 636-2 code, then set $\backslash\text{CurrentTrackedIsoCode}$ to the ISO 636-2 code and go back to step 8.

- 10. If there is an ISO 3166-1 code, the $\langle tag \rangle$ ($\backslash\text{CurrentTrackedTag}$) is then set to $\backslash\text{CurrentTrackedRegion}$, and if the file $\langle prefix \rangle \langle tag \rangle \langle suffix \rangle$ exists, the condition evaluates to *true*.

- 11. Finally, the $\langle tag \rangle$ ($\backslash\text{CurrentTrackedTag}$) is set to the root language label and if the file $\langle prefix \rangle \langle tag \rangle \langle suffix \rangle$ exist the condition evaluates to *true* otherwise it evaluates to *false*.

The $\backslash\text{IfTrackedLanguageFileExists}$ command sets the following commands which may be used within $\langle true\ part \rangle$ or $\langle false\ part \rangle$:

$\backslash\text{CurrentTrackedDialect}$

This command will always be set to the first argument of $\backslash\text{IfTrackedLanguageFileExists}$.

$\backslash\text{CurrentTrackedTag}$

This command will be set to $\langle tag \rangle$ if the file $\langle prefix \rangle \langle tag \rangle \langle suffix \rangle$ is found. If this command is empty, then the dialect hasn't been detected. If the dialect has been detected, but no file can be found, then $\backslash\text{CurrentTrackedTag}$ is set to the final attempt at determining $\langle tag \rangle$.

`\CurrentTrackedLanguage`

If the dialect hasn't been detected, this command will be empty, otherwise it will expand to the root language label (which may be the same as the dialect label).

`\CurrentTrackedRegion`

If the dialect hasn't been detected, this command will be empty. If the dialect has been assigned an ISO 3166-1 code, `\CurrentTrackedRegion` will expand to that code, otherwise it will be empty.

`\CurrentTrackedIsoCode`

If the dialect hasn't been detected, this command will be empty. Otherwise it may be empty or it may expand to either the ISO 639-1 or ISO 639-2 code (see above).

For example:

```
\AnyTrackedLanguages
{%
  \ForEachTrackedDialect{\ThisDialect}%
  {% try to load the language file for this dialect
    \IfTrackedLanguageFileExists{\ThisDialect}%
    {mypackage-}% file prefix
    {.ldf}% file suffix
    {\input{mypackage-\CurrentTrackedTag.ldf}}% file found
    {% not found
      \PackageWarning{mypackage}{No support for language
        '\ThisDialect'}%
    }%
  }%
}%
}
{% no languages detected so use defaults
}
```

If `\ThisDialect` is `british`, then the file search will be: `mypackage-british.ldf`, `mypackage-en-GB.ldf`, `mypackage-eng-GB.ldf`, `mypackage-en.ldf`, `mypackage-eng.ldf`, `mypackage-GB.ldf`, `mypackage-english.ldf`.

If `\ThisDialect` is `francais`, then the file search will be: `mypackage-francais.ldf`, `mypackage-fr.ldf`, `mypackage-fra.ldf`, `mypackage-french.ldf`.

Alternatively, if you're only providing support for the root languages:

```
\AnyTrackedLanguages
{%
  \ForEachTrackedLanguage{\ThisLanguage}%
  {% try to load the language file for this root language
    \IfTrackedLanguageFileExists{\ThisLanguage}%
  }%
}
```

```

{mypackage-}% file prefix
{.ldf}% file suffix
{\input{mypackage-\CurrentTrackedTag.ldf}}% file found
{% not found
  \PackageWarning{mypackage}{No support for language
    '\ThisLanguage'}%
}%
}%
}
{% no languages detected so use defaults
}

```

2.1 Examples

The examples in this section illustrate the above commands.

2.1.1 animals.sty

This example is for a trivial package called `animals.sty` that defines three textual commands: `\catname`, `\dogname` and `\ladybirdname`. The default values are: “cat”, “dog” and “bishy-barney-bee”.¹

The supported languages are defined in files with the prefix `animals-` and the suffix `.ldf`.

Here’s the code for the `animals.sty`:

```

\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{animals}

\RequirePackage{tracklang}

\newcommand\catname{cat}
\newcommand\dogname{dog}
\newcommand\ladybirdname{bishy-barney-bee}

\AnyTrackedLanguages
{%
  \ForEachTrackedDialect{\this@dialect}{%
    \IfTrackedLanguageFileExists{\this@dialect}%
    {animals-}% prefix
    {.ldf}%
    {%
      \PackageInfo{tracklang}{Loading ‘animals-\CurrentTrackedTag.ldf’
        for language ‘\this@dialect’}%
      \input{animals-\CurrentTrackedTag.ldf}%
    }%
  }%
  {%
    \PackageWarning{animals}%
  }%
}

```

¹Thass Broad Norfolk, my bewties :-P

```

        {No support for language '\this@dialect'}%
    }%
}%
}
{% no tracked languages, default already set up
}

```

\endinput

The animals-english.1df file contains:

```

\providecommand*\englishanimals{%
  \renewcommand*\catname{cat}%
  \renewcommand*\dogname{dog}%
  \renewcommand*\ladybirdname{bishy-barney-bee}%
}
\englishanimals

```

The animals-en-GB.1df file contains:

```

\input{animals-english.1df}

\providecommand*\englishGBanimals{%
  \englishanimals
  \renewcommand*\ladybirdname{ladybird}%
}
\englishGBanimals

```

The animals-en-US.1df file contains:

```

\input{animals-english.1df}

\providecommand*\englishUSanimals{%
  \englishanimals
  \renewcommand*\ladybirdname{ladybug}%
}
\englishUSanimals

```

This simple example doesn't try to use babel or polyglossia's \captions<*language*> mechanism, which means that if any users need to switch language part way through the document they will have to use \englishanimals, \englishGBanimals or \englishUSanimals to redefine the fixed names provided by this package.

The other drawback to this package is that if the user requests both british (or UKenglish) and american (or USenglish), then the animals-english.1df file is loaded twice, once by animals-en-GB.1df and once by animals-en-US.1df. This redundant input can be avoided by implementing a mechanism similar to \RequirePackage. The example package below, animals2.sty adds the following definitions:

```

\newcommand*\RequireAnimalsLang}[1]{%

```

```

\@ifundefined{ver@animals2-#1.ldf}{\input{animals2-#1.ldf}}{}%
}

\newcommand*{\ProvidesAnimalsLang}[1]{%
  \ProvidesFile{animals2-#1.ldf}%
}

```

Each .ldf file should now identify itself with \ProvidesAnimalsLang so it can then be loaded with \RequireAnimalsLang. The complete code for animals2.sty is now:

```

\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{animals2}

\RequirePackage{tracklang}

\newcommand\catname{cat}
\newcommand\dogname{dog}
\newcommand\ladybirdname{bishy-barney-bee}

\newcommand*{\RequireAnimalsLang}[1]{%
  \@ifundefined{ver@animals2-#1.ldf}{\input{animals2-#1.ldf}}{}%
}

\newcommand*{\ProvidesAnimalsLang}[1]{%
  \ProvidesFile{animals2-#1.ldf}%
}

\AnyTrackedLanguages
{%
  \ForEachTrackedDialect{\this@dialect}{%
    \IfTrackedLanguageFileExists{\this@dialect}%
    {animals2-}% prefix
    {.ldf}%
    {%
      \RequireAnimalsLang{\CurrentTrackedTag}%
    }%
    {%
      \PackageWarning{animals2}%
      {No support for language ‘\this@dialect’}%
    }%
  }%
}
{% no tracked languages, default already set up
}

\endinput

```

The animals2-english.ldf file is now:

```

\ProvidesAnimalsLang{english}

\providecommand*\englishanimals{%
  \renewcommand*\catname{cat}%
  \renewcommand*\dogname{dog}%
  \renewcommand*\ladybirdname{bishy-barney-bee}%
}
\englishanimals

```

The animals2-en-GB. ldf file is now:

```

\ProvidesAnimalsLang{en-GB}

\RequireAnimalsLang{english}%

\providecommand*\englishGBanimals{%
  \englishanimals
  \renewcommand*\ladybirdname{ladybird}%
}
\englishGBanimals

```

The animals2-en-US. ldf file is now:

```

\ProvidesAnimalsLang{en-US}

\RequireAnimalsLang{english}%

\providecommand*\englishUSanimals{%
  \englishanimals
  \renewcommand*\ladybirdname{ladybug}%
}

\englishUSanimals

```

Here's a sample document:

```

\documentclass[british,american,a4paper]{article}

\usepackage{animals2}

\begin{document}

\englishGBanimals

\catname.
\dogname.
\ladybirdname.

\englishUSanimals

\catname.

```

```

\dogname.
\ladybirdname.

\end{document}

```

The transcript includes:

```

(./animals2.sty
(./tracklang.sty) (./animals2-en-US.ldf (./animals2-english.ldf))
(./animals2-en-GB.ldf))

```

which shows that `animals2-english.ldf` is only input once.

The next step is to integrate with `babel` or `polyglossia`. Both these packages use `\captions⟨language⟩` to update fixed names whenever the given `⟨language⟩` is selected. With `babel` the `⟨language⟩` setting is the dialect label (which may also be the root language label). With `polyglossia` `⟨language⟩` is the root language label, regardless of the selected variant.

For example, if you load `babel` using:

```
\usepackage[UKenglish]{babel}
```

then `\captionsUKenglish` is defined but if you use:

```
\usepackage[british]{babel}
```

then `\captionsbritish` is defined instead. If, however, you load `polyglossia` using:

```

\usepackage{polyglossia}
\setmainlanguage[variant=british]{english}

```

then `\captionsenglish` is defined.

This just requires a few minor modifications to the above example package. Remember that the `.ldf` files are loaded within `\IfTrackedLanguageFileExists`, which means that the dialect label can be obtained from `\CurrentTrackedDialect`. To allow for the possibility of loading a language file at some later date, `\IfTrackedLanguageFile` can be moved to a new user command:

```

\newcommand*{\RequireAnimalsDialect}[1]{%
  \IfTrackedLanguageFileExists{#1}%
  {animals3-}% prefix
  {.ldf}%
  {%
    \RequireAnimalsLang{\CurrentTrackedTag}%
  }%
  {%
    \PackageWarning{animals3}%
    {No support for language ‘#1’}%
  }%
}

```

Now animals3-en-GB.1df and animals3-en-US.1df can use \CurrentTrackedDialect to add to the language captions mechanism. The file animals3-en-GB.1df contains:

```
\ProvidesAnimalsLang{en-GB}

\RequireAnimalsLang{english}%

\providecommand*\englishGBanimals{%
  \englishanimals
  \renewcommand*\ladybirdname{ladybird}%
}

\englishGBanimals

\@ifundefined{captions\CurrentTrackedDialect}%
{%
  \@ifundefined{captions\CurrentTrackedLanguage}{}%
  {%
    \expandafter\g@addto@macro
      \csname captions\CurrentTrackedLanguage\endcsname{\englishGBanimals}%
  }
}%
{%
  \expandafter\g@addto@macro
    \csname captions\CurrentTrackedDialect\endcsname{\englishGBanimals}%
}
```

The file animals3-en-GB.1df contains:

```
\ProvidesAnimalsLang{en-US}

\RequireAnimalsLang{english}%

\providecommand*\englishUSanimals{%
  \englishanimals
  \renewcommand*\ladybirdname{ladybug}%
}

\englishUSanimals

\@ifundefined{captions\CurrentTrackedDialect}%
{%
  \@ifundefined{captions\CurrentTrackedLanguage}{}%
  {%
    \expandafter\g@addto@macro
      \csname captions\CurrentTrackedLanguage\endcsname{\englishUSanimals}%
  }
}%
{%
```

```

\expandafter\g@addto@macro
\csname captions\CurrentTrackedDialect\endcsname{\englishUSanimals}%
}

```

The inner `\@ifundefined` check is required in case polyglossia has been loaded, although this causes some redundancy as `\captionseenglish` now contains both `\englishanimals` and `\englishGBanimals` (or `\englishUSanimals`).

The root language file `animals3-english.ldf` contains:

```

\ProvidesAnimalsLang{english}

\providecommand*\englishanimals{%
  \renewcommand*\catname{cat}%
  \renewcommand*\dogname{dog}%
  \renewcommand*\ladybirdname{bishy-barney-bee}%
}

\englishanimals

\@ifundefined{captionseenglish}%
{%
  \@ifundefined{captions\CurrentTrackedDialect}{}%
  {%
    \expandafter\g@addto@macro
    \csname captions\CurrentTrackedDialect\endcsname{\englishanimals}%
  }
}%
{%
  \g@addto@macro\captionseenglish{\englishanimals}%
}

```

The additional `\@ifundefined` check ensures that the package will work with other English dialects (although it may not produce the correct translations for the given region). You'll have to decide on the most appropriate default translations for the root language.

The complete code for the `animals3.sty` package is as follows:

```

\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{animals3}

\RequirePackage{tracklang}

\newcommand\catname{cat}
\newcommand\dogname{dog}
\newcommand\ladybirdname{bishy-barney-bee}

\newcommand*\RequireAnimalsDialect}[1]{%
  \IfTrackedLanguageFileExists{#1}%
  {animals3-}% prefix
  {.ldf}%
}

```



```

    {%
      \RequireAnimalsLang{\CurrentTrackedTag}%
    }%
    {%
      \PackageWarning{animals3}%
        {No support for language ‘#1’}%
    }%
  }

\newcommand*{\RequireAnimalsLang}[1]{%
  \@ifundefined{ver@animals3-#1.1df}{\input{animals3-#1.1df}}{}%
}

\newcommand*{\ProvidesAnimalsLang}[1]{%
  \ProvidesFile{animals3-#1.1df}%
}

\AnyTrackedLanguages
{%
  \ForEachTrackedDialect{\this@dialect}{%
    \RequireAnimalsDialect\this@dialect
  }%
}
{% no tracked languages, default already set up
}

\endinput

Here's sample document that uses this package with babel:

\documentclass[a4paper]{article}

\usepackage[british,american]{babel}

\usepackage{animals3}

\begin{document}

\selectlanguage{british}

\catname.
\dogname.
\ladybirdname.

\selectlanguage{american}

\catname.
\dogname.
\ladybirdname.

```

```
\end{document}
```

Unfortunately when used with polyglossia, tracklang can't detect the requested dialect. Consider the following example document:

```
\documentclass[a4paper]{article}

\usepackage{polyglossia}
\setmainlanguage[variant=british]{english}

\usepackage{animals3}

\begin{document}

\catname.
\dogname.
\ladybirdname.

\end{document}
```

This loads animals3-english.ldf but doesn't load animals3-en-GB.ldf. The dialect has to be added to the document class options:

```
\documentclass[british,a4paper]{article}

\usepackage{polyglossia}
\setmainlanguage[variant=british]{english}

\usepackage{animals3}

\begin{document}

\catname.
\dogname.
\ladybirdname.

\end{document}
```

Other languages can be added by simply providing additional .ldf files. For example, animals3-german.ldf:

```
\ProvidesAnimalsLang{german}

\providecommand*\germananimals{%
  \renewcommand*\catname{Katze}%
  \renewcommand*\dogname{Hund}%
  \renewcommand*\ladybirdname{Marienk\"afer}%
}

\germananimals
```

```

\@ifundefined{captionsgerman}%
{%
  \@ifundefined{captions\CurrentTrackedDialect}{-}%
  {%
    \expandafter\g@addto@macro
      \csname captions\CurrentTrackedDialect\endcsname{\germananimals}%
  }
}%
{%
  \g@addto@macro\captionsgerman{\germananimals}%
}

```

This works fine with babel:

```

\documentclass[a4paper]{article}

\usepackage[ngerman]{babel}

\usepackage{animals3}

\begin{document}

\catname.
\dogname.
\ladybirdname.

\end{document}

```

It also works with ngerman:

```

\documentclass[a4paper]{article}

\usepackage{ngerman}

\usepackage{animals3}

\begin{document}

\catname.
\dogname.
\ladybirdname.

\end{document}

```

and with translator:

```

\documentclass[a4paper]{article}

\usepackage[ngerman]{translator}

\usepackage{animals3}

```

```
\begin{document}
```

```
\catname.
```

```
\dogname.
```

```
\ladybirdname.
```

```
\end{document}
```

Since no regional variants have been provided to accompany `animals3-german.1df` this also works fine with `polyglossia`:

```
\documentclass[a4paper]{article}
```

```
\usepackage{polyglossia}
```

```
\setmainlanguage{german}
```

```
\usepackage{animals3}
```

```
\begin{document}
```

```
\catname.
```

```
\dogname.
```

```
\ladybirdname.
```

```
\end{document}
```

2.1.2 `regions.sty`

Earlier, I mentioned the search order for `\IfTrackedLanguageFileExists` where if, for example, the dialect is `british`, the file search will be: `mypackage-british.1df`, `mypackage-en-GB.1df`, `mypackage-eng-GB.1df`, `mypackage-en.1df`, `mypackage-eng.1df`, `mypackage-GB.1df`, `mypackage-english.1df`. You may have wondered why `mypackage-GB.1df` is included in the search. Since some countries have multiple official languages, the country code on its own may not indicate the language.

The reason for including just the country code as the *<tag>* in the file search is to allow for region rather than language dependent settings. For example, suppose I want to write a package that needs to know whether to use imperial or metric measurements in the document, but I also want to provide multilingual support. The language alone won't tell me whether to use imperial or metric (for example, the US uses imperial and the UK uses metric). I could provide `.1df` files for every language and region combination, but this would result in a lot of redundancy.

The example package (`regions.sty`) below illustrates this.

```
\NeedsTeXFormat{LaTeX2e}
```

```
\ProvidesPackage{regions}
```

```

\RequirePackage{tracklang}

\newcommand*{\weightunit}{kg}
\newcommand*{\lengthunit}{mm}
\newcommand*{\currencyunit}{EUR}

\newcommand*{\unitname}{units}

\AnyTrackedLanguages
{%
  \ForEachTrackedDialect{\this@dialect}{%
    \IfTrackedLanguageFileExists{\this@dialect}%
    {regions-}% prefix
    {.ldf}%
    {%
      \PackageInfo{tracklang}{Loading ‘regions-\CurrentTrackedTag.ldf’
        for language ‘\this@dialect’}%
      \input{regions-\CurrentTrackedTag.ldf}%
      \InputIfFileExists{regions-\CurrentTrackedLanguage.ldf}%
      {}%
      {%
        \PackageWarning{regions}{No support for language
          ‘\CurrentTrackedLanguage’}%
      }%
    }%
  }%
  {%
    \PackageWarning{regions}%
    {No support for language ‘\this@dialect’}%
  }%
}%
}
{% no tracked languages, default already set up
}

\endinput

```

There are separate .ldf files for region and language. First are the regions.

- regions-BE.ldf (Belgium):

```

\providecommand*{\BEunits}{%
  \renewcommand*{\weightunit}{kg}%
  \renewcommand*{\lengthunit}{mm}%
  \renewcommand*{\currencyunit}{EUR}%
}

\BEunits

```

- regions-CA.ldf (Canada):

```

\providecommand*\CAunits{%
  \renewcommand*\weightunit\{kg\}%
  \renewcommand*\lengthunit\{mm\}%
  \renewcommand*\currencyunit\{CAD\}%
}

```

\CAunits

- regions-GB.ldf (Great Britain):

```

\providecommand*\GBunits{%
  \renewcommand*\weightunit\{kg\}%
  \renewcommand*\lengthunit\{mm\}%
  \renewcommand*\currencyunit\{GBP\}%
}

```

\GBunits

- regions-US.ldf (USA):

```

\providecommand*\USunits{%
  \renewcommand*\weightunit\{lb\}%
  \renewcommand*\lengthunit\{in\}%
  \renewcommand*\currencyunit\{USD\}%
}

```

\USunits

Now follow the language files.

- regions-dutch.ldf:

```

\providecommand*\dutchnames{%
  \renewcommand*\unitname\{meeteenheden\}%
}

```

\dutchnames

- regions-english.ldf:

```

\providecommand*\englishnames{%
  \renewcommand*\unitname\{units\}%
}

```

\englishnames

- `regions-french.ldf`:

```
\providecommand*\frenchnames{%
  \renewcommand*\unitname{\unit\'es}%
}

\frenchnames
```

- `regions-german.ldf`:

```
\providecommand*\germannames{%
  \renewcommand*\unitname{\Ma\ss einheiten}%
}

\germannames
```

Here's an example document that uses this package:

```
\documentclass[canadien]{article}

\usepackage{regions}

\begin{document}

\unitname: \weightunit, \lengthunit, \currencyunit.

\end{document}
```

This works because `\IfTrackedLanguageFileExists` searches for the country code before the root language label. In this case, `regions-CA.ldf` is matched rather than `regions-french.ldf`, so `regions-CA.ldf` is loaded by

```
\input{regions-\CurrentTrackedTag.ldf}%
```

After this, the language file `regions-french.ldf` is then loaded:

```
\InputIfFileExists{regions-\CurrentTrackedLanguage.ldf}%
{}%
{%
  \PackageWarning{regions}{No support for language
    '\CurrentTrackedLanguage'}%
}%
```

This assumes that there's a country code `.ldf` file available. This example needs a little modification to use default units in case the region is missing:

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{regions}

\RequirePackage{tracklang}
```

```

\newcommand*{\weightunit}{kg}
\newcommand*{\lengthunit}{mm}
\newcommand*{\currencyunit}{EUR}

\newcommand*{\unitname}{units}

\newcommand*{\defaultunits}{%
  \renewcommand*{\weightunit}{kg}%
  \renewcommand*{\lengthunit}{mm}%
  \renewcommand*{\currencyunit}{EUR}%
}

\AnyTrackedLanguages
{%
  \ForEachTrackedDialect{\this@dialect}{%
    \IfTrackedLanguageFileExists{\this@dialect}%
    {regions-}% prefix
    {.ldf}%
    {%
      \PackageInfo{tracklang}{Loading ‘regions-\CurrentTrackedTag.ldf’
        for language ‘\this@dialect’}%
      \input{regions-\CurrentTrackedTag.ldf}%
      \ifx\CurrentTrackedTag\CurrentTrackedLanguage
        \defaultunits
      \else
        \InputIfFileExists{regions-\CurrentTrackedLanguage.ldf}%
        {}%
        {%
          \PackageWarning{regions}{No support for language
            ‘\CurrentTrackedLanguage’}%
        }%
      \fi
    }%
  }%
  {%
    \PackageWarning{regions}%
    {No support for language ‘\this@dialect’}%
  }%
}%
}
{% no tracked languages, default already set up
}

\endinput

```

As with the trivial example in the previous section, this package hasn't hooked into 's caption mechanism, but this can be implemented for this example in a similar manner to the final example package in the previous section.

3 Adding Support for Language Tracking

If you are writing a package that *defines* languages (rather than a package that provides multilingual support if the user has already loaded a language package) then you can load tracklang and use the commands below to help other packages track your defined languages.

You can notify tracklang that a user has requested a particular dialect using:

```
\AddTrackedDialect{<dialect>}{<root language label>}
```

where *<dialect>* is the dialect label and *<root language label>* is the label for the dialect's root language. This command may also be used for synonyms. If the dialect is already in the tracked dialect list, it won't be added again. If the root language is already in the tracked language list, it won't be added again.

```
\AddTrackedLanguage{<root language label>}
```

This is equivalent to `\AddTrackedDialect{<root language label>}{<root language label>}` and is provided for the cases where the language doesn't have any variants or the user wants the default variant for that language.

The ISO codes are specified using:

```
\AddTrackedIsoLanguage{<code type>}{<code>}{<label>}
```

where *<code type>* is the ISO code type (for example, 639-1), *<code>* is the code (for example, "en") and *<label>* is the language or dialect label. (The root language label for ISO 639-1 or ISO 639-2 and the dialect label for the country code ISO 3166-1.

For example:

```
\ProvidesPackage{alien}

\RequirePackage{tracklang}

\DeclareOption{martian}{%
  \AddTrackedLanguage{martian}%
  \AddTrackedIsoLanguage{639-1}{xx}{martian}%
  % code to set Martian language
}

\DeclareOption{uppermartian}{%
  \AddTrackedDialect{uppermartian}{martian}%
  \AddTrackedIsoLanguage{639-1}{xx}{martian}%
  \AddTrackedIsoLanguage{3166-1}{XX}{uppermartian}%
  % code to set Upper Martian dialect
}
```

```

\DeclareOption{lowermartian}{%
  \AddTrackedDialect{lowermartian}{martian}%
  \AddTrackedIsoLanguage{639-1}{xx}{martian}%
  \AddTrackedIsoLanguage{3166-1}{YY}{lowermartian}%
  % code to set Lower Martian dialect
}

```

Now other package writers who want to provide support for the Martian dialects can easily detect which language options the user requested through your package, and it means the user doesn't have to specify all their language options through the document class or through each package that has multilingual support.

If you want to track one of the dialects known to tracklang (for example, "british") you can do so using:

```
\TrackPredefinedDialect{<dialect label>}
```

This will add the dialect and the ISO codes.

For non-LaTeX packages, you can input tracklang.tex provided the category code for @ has been set to 11:

```

\catcode '@11\relax
\input tracklang

```

(Remember to revert it back afterwards if necessary.)

4 Plain T_EX Example

Since plain T_EX doesn't have document class or package options, the dialects and ISO codes need to be added using the commands described in the previous section. For example:

```

\catcode '@11\relax
\input tracklang
\catcode '@12\relax

```

```

\TrackPredefinedDialect{british}
\TrackPredefinedDialect{american}
\TrackPredefinedDialect{canadian}
\TrackPredefinedDialect{canadien}
\TrackPredefinedDialect{belgique}

```

Dialect Regions:

```

\ForEachTrackedDialect{\ThisDialect}{\ThisDialect:
  \TrackedIsoCodeFromLanguage{\TwoLetterIsoCountryCode}{\ThisDialect}. }

```

Language Codes:

```
\ForEachTrackedLanguage{\ThisLanguage}{\ThisLanguage:  
  \TrackedIsoCodeFromLanguage{\TwoLetterIsoLanguageCode}{\ThisLanguage}. }
```

\bye

This produces:

Dialect Regions:

british: GB. american: US. canadian: CA. canadien: CA. bel-
gique: BE.

Language Codes:

english: en. french: fr.

Found “animals-en-GB.ldf”

Found “animals-en-US.ldf”

Found “animals-english.ldf”

No file found for dialect “canadien”

No file found for dialect “belgique”

5 The Code

5.1 L^AT_EX Code (tracklang.sty)

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{tracklang}[2014/09/29 v1.0 (NLCT) Track Languages]
```

Load generic code:

```
3 \input{tracklang}
```

Define package/class options:

```
4 \DeclareOption{acadian}{\@tracklang@add@acadian}
5 \DeclareOption{afrikaans}{\@tracklang@add@afrikaans}
6 \DeclareOption{albanian}{\@tracklang@add@albanian}
7 \DeclareOption{american}{\@tracklang@add@american}
8 \DeclareOption{amharic}{\@tracklang@add@amharic}
9 \DeclareOption{anglosaxon}{\@tracklang@add@anglosaxon}
10 \DeclareOption{apache}{\@tracklang@add@apache}
11 \DeclareOption{arabic}{\@tracklang@add@arabic}
12 \DeclareOption{armenian}{\@tracklang@add@armenian}
13 \DeclareOption{asturian}{\@tracklang@add@asturian}
14 \DeclareOption{australian}{\@tracklang@add@australian}
15 \DeclareOption{austrian}{\@tracklang@add@austrian}
16 \DeclareOption{bahasa}{\@tracklang@add@bahasa}
17 \DeclareOption{bahasai}{\@tracklang@add@bahasai}
18 \DeclareOption{bahasam}{\@tracklang@add@bahasam}
19 \DeclareOption{basque}{\@tracklang@add@basque}
20 \DeclareOption{belgique}{\@tracklang@add@belgique}
21 \DeclareOption{bengali}{\@tracklang@add@bengali}
22 \DeclareOption{brazil}{\@tracklang@add@brazil}
23 \DeclareOption{brazilian}{\@tracklang@add@brazilian}
24 \DeclareOption{breton}{\@tracklang@add@breton}
25 \DeclareOption{british}{\@tracklang@add@british}
26 \DeclareOption{bulgarian}{\@tracklang@add@bulgarian}
27 \DeclareOption{canadian}{\@tracklang@add@canadian}
28 \DeclareOption{canadien}{\@tracklang@add@canadien}
29 \DeclareOption{catalan}{\@tracklang@add@catalan}
30 \DeclareOption{coptic}{\@tracklang@add@coptic}
31 \DeclareOption{croatian}{\@tracklang@add@croatian}
32 \DeclareOption{cymraeg}{\@tracklang@add@cymraeg}
33 \DeclareOption{czech}{\@tracklang@add@czech}
34 \DeclareOption{danish}{\@tracklang@add@danish}
35 \DeclareOption{deutsch}{\@tracklang@add@deutsch}
36 \DeclareOption{divehi}{\@tracklang@add@divehi}
37 \DeclareOption{dutch}{\@tracklang@add@dutch}
38 \DeclareOption{english}{\@tracklang@add@english}
39 \DeclareOption{esperanto}{\@tracklang@add@esperanto}
40 \DeclareOption{estonian}{\@tracklang@add@estonian}
41 \DeclareOption{farsi}{\@tracklang@add@farsi}
42 \DeclareOption{finnish}{\@tracklang@add@finnish}
43 \DeclareOption{flemish}{\@tracklang@add@flemish}
```

```

44 \DeclareOption{francais}{\@tracklang@add@francais}
45 \DeclareOption{french}{\@tracklang@add@french}
46 \DeclareOption{frenchb}{\@tracklang@add@frenchb}
47 \DeclareOption{friulan}{\@tracklang@add@friulan}
48 \DeclareOption{friulano}{\@tracklang@add@friulano}
49 \DeclareOption{friulian}{\@tracklang@add@friulian}
50 \DeclareOption{gaeilge}{\@tracklang@add@gaeilge}
51 \DeclareOption{galician}{\@tracklang@add@galician}
52 \DeclareOption{galicien}{\@tracklang@add@galicien}
53 \DeclareOption{german}{\@tracklang@add@german}
54 \DeclareOption{greek}{\@tracklang@add@greek}
55 \DeclareOption{hebrew}{\@tracklang@add@hebrew}
56 \DeclareOption{hindi}{\@tracklang@add@hindi}
57 \DeclareOption{hungarian}{\@tracklang@add@hungarian}
58 \DeclareOption{icelandic}{\@tracklang@add@icelandic}
59 \DeclareOption{indon}{\@tracklang@add@indon}
60 \DeclareOption{indonesian}{\@tracklang@add@indonesian}
61 \DeclareOption{interlingua}{\@tracklang@add@interlingua}
62 \DeclareOption{irish}{\@tracklang@add@irish}
63 \DeclareOption{italian}{\@tracklang@add@italian}
64 \DeclareOption{kannada}{\@tracklang@add@kannada}
65 \DeclareOption{lao}{\@tracklang@add@lao}
66 \DeclareOption{latin}{\@tracklang@add@latin}
67 \DeclareOption{latein}{\@tracklang@add@latein}
68 \DeclareOption{latvian}{\@tracklang@add@latvian}
69 \DeclareOption{lithuanian}{\@tracklang@add@lithuanian}
70 \DeclareOption{lsorbian}{\@tracklang@add@lsorbian}
71 \DeclareOption{magyar}{\@tracklang@add@magyar}
72 \DeclareOption{malay}{\@tracklang@add@malay}
73 \DeclareOption{malayalam}{\@tracklang@add@malayalam}
74 \DeclareOption{marathi}{\@tracklang@add@marathi}
75 \DeclareOption{meyalu}{\@tracklang@add@meyalu}
76 \DeclareOption{naustrian}{\@tracklang@add@naustrian}
77 \DeclareOption{ngerman}{\@tracklang@add@ngerman}
78 \DeclareOption{lowersorbian}{\@tracklang@add@lowersorbian}
79 \DeclareOption{newzealand}{\@tracklang@add@newzealand}
80 \DeclareOption{nko}{\@tracklang@add@nko}
81 \DeclareOption{norsk}{\@tracklang@add@norsk}
82 \DeclareOption{norwegian}{\@tracklang@add@norwegian}
83 \DeclareOption{nynorsk}{\@tracklang@add@nynorsk}
84 \DeclareOption{occitan}{\@tracklang@add@occitan}
85 \DeclareOption{persian}{\@tracklang@add@persian}
86 \DeclareOption{piedmontese}{\@tracklang@add@piedmontese}
87 \DeclareOption{piemonteis}{\@tracklang@add@piemonteis}
88 \DeclareOption{polish}{\@tracklang@add@polish}
89 \DeclareOption{polutoniko}{\@tracklang@add@polutoniko}
90 \DeclareOption{polutonikogreek}{\@tracklang@add@polutonikogreek}
91 \DeclareOption{portuges}{\@tracklang@add@portuges}
92 \DeclareOption{portuguese}{\@tracklang@add@portuguese}

```

```

93 \DeclareOption{romanche}{\@tracklang@add@romanche}
94 \DeclareOption{romanian}{\@tracklang@add@romanian}
95 \DeclareOption{romansh}{\@tracklang@add@romansh}
96 \DeclareOption{romansch}{\@tracklang@add@romansch}
97 \DeclareOption{rumantsch}{\@tracklang@add@rumantsch}
98 \DeclareOption{russian}{\@tracklang@add@russian}
99 \DeclareOption{russianb}{\@tracklang@add@russianb}
100 \DeclareOption{samin}{\@tracklang@add@samin}
101 \DeclareOption{sanskrit}{\@tracklang@add@sanskrit}
102 \DeclareOption{scottish}{\@tracklang@add@scottish}
103 \DeclareOption{gaelic}{\@tracklang@add@gaelic}
104 \DeclareOption{serbian}{\@tracklang@add@serbian}
105 \DeclareOption{slovak}{\@tracklang@add@slovak}
106 \DeclareOption{slovene}{\@tracklang@add@slovene}
107 \DeclareOption{slovenian}{\@tracklang@add@slovenian}
108 \DeclareOption{spanish}{\@tracklang@add@spanish}
109 \DeclareOption{swedish}{\@tracklang@add@swedish}
110 \DeclareOption{syriac}{\@tracklang@add@syriac}
111 \DeclareOption{tamil}{\@tracklang@add@tamil}
112 \DeclareOption{telugu}{\@tracklang@add@telugu}
113 \DeclareOption{tai}{\@tracklang@add@tai}
114 \DeclareOption{thai}{\@tracklang@add@thai}
115 \DeclareOption{tibetan}{\@tracklang@add@tibetan}
116 \DeclareOption{turkish}{\@tracklang@add@turkish}
117 \DeclareOption{turkmen}{\@tracklang@add@turkmen}
118 \DeclareOption{UKenglish}{\@tracklang@add@UKenglish}
119 \DeclareOption{ukrainian}{\@tracklang@add@ukrainian}
120 \DeclareOption{ukraine}{\@tracklang@add@ukraine}
121 \DeclareOption{ukraineb}{\@tracklang@add@ukraine}
122 \DeclareOption{urdu}{\@tracklang@add@urdu}
123 \DeclareOption{usorbian}{\@tracklang@add@usorbian}
124 \DeclareOption{uppersorbian}{\@tracklang@add@uppersorbian}
125 \DeclareOption{USenglish}{\@tracklang@add@USenglish}
126 \DeclareOption{valencian}{\@tracklang@add@valencian}
127 \DeclareOption{valencien}{\@tracklang@add@valencien}
128 \DeclareOption{vietnamese}{\@tracklang@add@vietnamese}
129 \DeclareOption{walloon}{\@tracklang@add@walloon}
130 \DeclareOption{welsh}{\@tracklang@add@welsh}
131 \let\@tracklang@declaredoptions\@declaredoptions
132 \ProcessOptions

```

In the event that the language hasn't been supplied through the package options (or through the class options, which the package options should process) we need to check if any of the known language packages have been loaded. This is a bit dangerous as it relies on the packages not changing their internal language macros. It would be easier if all the language packages could provide a reliable user interface to determine which languages have been loaded.

```

133 \ifx\@tracklang@languages\@empty

```

First try babel. If babel has been loaded, the languages are stored in \bbl@loaded, so check if this command has been defined, and if it has add those languages.

```
134 \ifundefined{bbl@loaded}%
135 {%
```

If translator has been loaded, the languages are stored in \trans@languages

```
136 \ifundefined{trans@languages}
137 {%
```

Has ngerman been loaded?

```
138 \ifpackageloaded{ngerman}%
139 {%
140 \@tracklang@add@ngerman
141 }%
142 {%
```

Has polyglossia been loaded?

```
143 \ifpackageloaded{polyglossia}
144 {%
```

polyglossia sets \<lang>@loaded for each loaded language, so check this for all known languages. I don't know how to check for variants.

```
145 \@for\this@language:=albanian,amharic,arabic,armenian,asturian,%
146 bahasai,bahasam,basque,bengali,brazil,breton,bulgarian,%
147 catalan,coptic,croatian,czech,danish,divehi,dutch,english,%
148 esperanto,estonian,farsi,finnish,french,friulan,galician,%
149 german,greek,hebrew,hindi,icelandic,interlingua,irish,%
150 italian,kannada,lao,latin,latvian,lithuanian,lsorbian,%
151 magyar,malayalam,marathi,nko,norsk,nynorsk,occitan,%
152 piedmontese,polish,portuges,romanian,romansh,russian,%
153 samin,sanskrit,scottish,serbian,slovak,slovenian,spanish,%
154 swedish,syriac,tamil,telugu,thai,tibetan,turkish,turkmen,%
155 ukrainian,urdu,usorbian,vietnamese,welsh\do{%
156 \ifundefined{\this@language @loaded}%
157 {}%
158 {%
159 \ifundefined{@tracklang@add@\this@language}%
160 {}%
161 \AddTrackedLanguage\this@language
162 }%
163 {%
164 \@nameuse{@tracklang@add@\this@language}%
165 }%
166 }%
167 }%
168 }%
169 {%
```

None of the known packages have been loaded, so do nothing in case another package wants to load this one before setting up the language options. However, if at this point babel has been loaded, then it's an older version that hasn't defined \bbl@loaded so check for this.

```

170         \@ifpackageloaded{babel}
171     {%
172         \PackageInfo{tracklang}{babel loaded but
173         \string\bb@loaded\space not defined. Will attempt
174         to track known languages.}%
175         \@for\this@language:=\@tracklang@declaredoptions\do{%
176             \@ifundefined{captions\this@language}%
177             {}%
178             {\@nameuse{@tracklang@add@\this@language}}}%
179         }%
180     }%
181     {}%
182 }%
183 }%
184 }%
185 {%

```

Add from translator. If translator has been loaded, the language names are stored in \trans@languages but these are labels used by translator, so some mapping is required.

```

186     \let\@tracklang@add@Acadian\@tracklang@add@acadian
187     \let\@tracklang@add@French\@tracklang@add@french
188     \let\@tracklang@add@Afrikaans\@tracklang@add@afrikaans
189     \let\@tracklang@add@Dutch\@tracklang@add@dutch
190     \let\@tracklang@add@AmericanEnglish\@tracklang@add@american
191     \let\@tracklang@add@Austrian\@tracklang@add@austrian
192     \@namedef{@tracklang@add@Austrian1997}{\@tracklang@add@naustrian}
193     \let\@tracklang@add@Bahasa\@tracklang@add@bahasa
194     \let\@tracklang@add@Basque\@tracklang@add@basque
195     \let\@tracklang@add@Brazilian\@tracklang@add@brazil
196     \let\@tracklang@add@Portuguese\@tracklang@add@portuguese
197     \let\@tracklang@add@Breton\@tracklang@add@breton
198     \let\@tracklang@add@BritishEnglish\@tracklang@add@british
199     \let\@tracklang@add@Bulgarian\@tracklang@add@bulgarian
200     \let\@tracklang@add@Canadian\@tracklang@add@canadian
201     \let\@tracklang@add@Canadien\@tracklang@add@canadien
202     \let\@tracklang@add@Catalan\@tracklang@add@catalan
203     \let\@tracklang@add@Croatian\@tracklang@add@croatian
204     \let\@tracklang@add@Czech\@tracklang@add@czech
205     \let\@tracklang@add@Danish\@tracklang@add@danish
206     \let\@tracklang@add@Dutch\@tracklang@add@dutch
207     \let\@tracklang@add@English\@tracklang@add@english
208     \let\@tracklang@add@Esperanto\@tracklang@add@esperanto
209     \let\@tracklang@add@Estonian\@tracklang@add@estonian
210     \let\@tracklang@add@Finnish\@tracklang@add@finnish
211     \let\@tracklang@add@French\@tracklang@add@french
212     \let\@tracklang@add@Galician\@tracklang@add@galician
213     \let\@tracklang@add@German\@tracklang@add@german
214     \@namedef{@tracklang@add@German1997}{\@tracklang@add@ngerman}
215     \let\@tracklang@add@Greek\@tracklang@add@greek

```



```

216 \let\@tracklang@add@Polutoniko\@tracklang@add@polutoniko
217 \let\@tracklang@add@Hebrew\@tracklang@add@hebrew
218 \let\@tracklang@add@Hungarian\@tracklang@add@hungarian
219 \let\@tracklang@add@Icelandic\@tracklang@add@icelandic
220 \let\@tracklang@add@Irish\@tracklang@add@irish
221 \let\@tracklang@add@Italian\@tracklang@add@italian
222 \let\@tracklang@add@Latin\@tracklang@add@latin
223 \let\@tracklang@add@LowerSorbian\@tracklang@add@lowersorbian
224 \let\@tracklang@add@Magyar\@tracklang@add@magyar
225 \let\@tracklang@add@Nynorsk\@tracklang@add@nynorsk
226 \let\@tracklang@add@Norsk\@tracklang@add@norsk
227 \let\@tracklang@add@Polish\@tracklang@add@polish
228 \let\@tracklang@add@Portuguese\@tracklang@add@portuguese
229 \let\@tracklang@add@Romanian\@tracklang@add@romanian
230 \let\@tracklang@add@Russian\@tracklang@add@russian
231 \let\@tracklang@add@Scottish\@tracklang@add@scottish
232 \let\@tracklang@add@Serbian\@tracklang@add@serbian
233 \let\@tracklang@add@Slovak\@tracklang@add@slovak
234 \let\@tracklang@add@Slovene\@tracklang@add@slovene
235 \let\@tracklang@add@Spanish\@tracklang@add@spanish
236 \let\@tracklang@add@Swedish\@tracklang@add@swedish
237 \let\@tracklang@add@Turkish\@tracklang@add@turkish
238 \let\@tracklang@add@Ukrainian\@tracklang@add@ukrainian
239 \let\@tracklang@add@UpperSorbian\@tracklang@add@uppersorbian
240 \let\@tracklang@add@Welsh\@tracklang@add@welsh

```

Now iterate through the declared languages:

```

241 \@for\this@language:=\trans@languages\do{%
242     \ifundefined{\@tracklang@add@\this@language}{}%
243     {\@nameuse{\@tracklang@add@\this@language}}%
244 }%
245 }%
246 }%
247 {%

```

Add from babel

```

248 \@for\this@language:=\bbl@loaded\do{%
249     \ifundefined{\@tracklang@add@\this@language}%
250     {%
251         \PackageWarning{tracklang}{Adding unknown language '\this@language'}%
252         \AddTrackedLangage{\this@language}%
253     }%
254     {\@nameuse{\@tracklang@add@\this@language}}%
255 }%
256 }

```

End of check for language packages

```
257 \fi
```

5.2 Generic Code (tracklang.tex)

First check if this file has already been loaded:

```

258 \ifx \@tracklang@languages\undefined
259 \else
260   \expandafter\endinput
261 \fi

```

Define a long command for determining the existence of a control sequence by its name.

```
\@tracklang@ifundef
```

```

262 \long\def\@tracklang@ifundef#1#2#3{%
263   \expandafter\ifx\csname #1\endcsname\relax
264     #2%
265   \else
266     #3%
267   \fi
268 }

```

Provide some commands in case the \LaTeX kernel hasn't been loaded.

```
\@tracklang@nnil
```

```

269 \ifx\@nnil\undefined
270   \def\@tracklang@nnil{\@nil}
271 \else
272   \let\@tracklang@nnil\@nnil
273 \fi

```

```
\@tracklang@for
```

```

274 \ifx\@for\undefined
275   \long\def\@tracklang@for#1:=#2\do#3{%
276     \expandafter\def\expandafter\@fortmp\expandafter{#2}%
277     \ifx\@fortmp\empty
278       \else
279         \expandafter
280           \@tracklang@forloop #2,\@nil,\@nil\@@ #1{#3}%
281       \fi
282   }
283 \long\def\@tracklang@forloop#1,#2,#3\@@ #4#5{%
284   \def #4{#1}%
285   \ifx#4\@tracklang@nnil
286   \else
287     #5%
288     \def #4{#2}%
289     \ifx#4\@tracklang@nnil
290     \else
291       #5%
292       \@tracklang@iforloop #3\@@ #4{#5}%
293     \fi
294   \fi
295 }
296 \long\def\@tracklang@iforloop#1,#2\@@ #3#4{%

```

```

297 \def#3{#1}%
298 \ifx#3\@tracklang@nnil
299 \expandafter
300 \@tracklang@fornoop
301 \else
302 #4\relax
303 \expandafter\@tracklang@iforloop
304 \fi
305 #2\@@ #3{#4}%
306 }
307 \long\def\@tracklang@fornoop#1\@@ #2#3{}
308 \else
309 \let\@tracklang@for\@for
310 \fi

```

\@tracklang@namedef

```

311 \ifx\@namedef\undefined
312 \def\@tracklang@namedef#1{\expandafter\def\csname#1\endcsname}
313 \else
314 \let\@tracklang@namedef\@namedef
315 \fi

```

\@tracklang@nameuse

```

316 \ifx\@nameuse\undefined
317 \def\@tracklang@nameuse#1{\csname#1\endcsname}
318 \else
319 \let\@tracklang@nameuse\@nameuse
320 \fi

```

\@tracklang@err

```

321 \ifx\PackageError\undefined
322 \def\@tracklang@err#1#2{%
323 \errehelp{#2}%
324 \errmessage{tracklang: #1}}
325 \else
326 \def\@tracklang@err#1#2{\PackageError{tracklang}{#1}{#2}}
327 \fi

```

cklang@ifFileExists

```

328 \ifx\IfFileExists\undefined
329 \long\def\@tracklang@ifFileExists#1#2#3{%
330 \openin0=#1 %
331 \ifeof0\relax
332 \def\reserved@a{#3}%
333 \else
334 \closein0\relax
335 \edef\@filef@und{#1 }%
336 \def\reserved@a{#2}%
337 \fi

```

```

338 \reserved@a
339 }
340
341 \else
342 \let\@tracklang@ifFileExists\IfFileExists
343 \fi

```

`\tracklang@languages` Provide a list to keep track of all the languages.

```

344 \def\@tracklang@languages{}

```

`\tracklang@dialects` Provide a list to keep track of all the dialects. Here the “dialect” isn’t necessarily an actual dialect but may be a root language or a synonym. It will usually correspond to the language name as specified by the user in the package option.

```

345 \def\@tracklang@dialects{}

```

`\@tracklang@ifinlist` `\@tracklang@ifinlist{<item>}{<list>}{<true part>}{<false part>}`

Checks if `<item>` is in `<list>`. (Performs a one-level expansion on `<list>` but no expansion on `<item>`.)

```

346 \def\@tracklang@ifinlist#1#2#3#4{%
347 \def\@tracklang@doifinlist##1,#1,##2\end@tracklang@doifinlist{%
348 \def\@before{##1}%
349 \def\@after{##2}%
350 }%
351 \expandafter\@tracklang@doifinlist\expandafter,#2,#1,\@nil
352 \end@tracklang@doifinlist
353 \ifx\@after\@tracklang@nnil
not found
354 #4%
355 \else
found
356 #3%
357 \fi
358 }

```

`\@tracklang@add` `\@tracklang@add{<item>}{<list cs>}`

Adds an item to the list given by `<list cs>`. Does nothing if `<item>` is empty or is already in the list. The `<item>` is fully expanded before being added.

```

359 \def\@tracklang@add#1#2{%
360 % \end{macrocode}
361 % First find out if the item is empty.
362 % \begin{macrocode}
363 \edef\@tracklang@element{#1}%
364 \ifx\@tracklang@element\empty

```

Item is empty, so do nothing.

```
365 \else
366 \expandafter\@tracklang@ifinlist\expandafter{\@tracklang@element}#2%
367 {%
```

Already in list, so do nothing.

```
368 }%
369 {%
```

Not in list, so add.

```
370 \ifx\empty#2\relax
371 \let#2\@tracklang@element
372 \else
373 \edef#2{#2,\@tracklang@element}%
374 \fi
375 }%
376 \fi
377 }
```

`\AddTrackedDialect` `\AddTrackedDialect{<dialect name>}{<language name>}`

Add a dialect.

```
378 \def\AddTrackedDialect#1#2{%
379 \@tracklang@add{#1}{\@tracklang@dialects}%
380 \@tracklang@add{#2}{\@tracklang@languages}%
381 \@tracklang@namedef{\@tracklang@fromdialect@#1}{#2}%
382 \@tracklang@ifundef{\@tracklang@todialect@#2}%
383 {\@tracklang@namedef{\@tracklang@todialect@#2}{#1}}%
384 {%
385 \def\@tracklang@lang{#1}%
386 \expandafter\@tracklang@add\expandafter\@tracklang@lang
387 \csname \@tracklang@todialect@#2\endcsname
388 }%
389 }
```

`\AddTrackedLanguage` `\AddTrackedLanguage{<language name>}`

Add a dialect.

```
390 \def\AddTrackedLanguage#1{%
391 \AddTrackedDialect{#1}{#1}%
392 }
```

`\AddTrackedIsoLanguage` `\AddTrackedIsoLanguage{<code type>}{<code>}{<language>}`

Adds a mapping between the given ISO code and language name. There may be multiple mappings from an ISO code to a language name, but only one mapping from a language name to an ISO code.

```

393 \def\AddTrackedIsoLanguage#1#2#3{%
394   \@tracklang@namedef{@tracklang@#1@isofromlang@#3}{#2}%
395   \@tracklang@ifundef{@tracklang@#1@isotolang@#2}%
396   {\@tracklang@namedef{@tracklang@#1@isotolang@#2}{#3}}%
397   {%
398     \def\@tracklang@lang{#3}%
399     \expandafter\@tracklang@add\expandafter\@tracklang@lang
400     \csname @tracklang@#1@isotolang@#2\endcsname
401   }%
402 }

```

dLanguageFromIsoCode

`\TrackedLanguageFromIsoCode{<code type>}{<code>}`

Fetches the language label (or labels) associated with the given code.

```

403 \def\TrackedLanguageFromIsoCode#1#2{%
404   \@tracklang@nameuse{@tracklang@#1@isotolang@#2}%
405 }

```

dIsoCodeFromLanguage

`\TrackedIsoCodeFromLanguage{<code type>}{<language>}`

Fetches the code associated with the given language or dialect.

```

406 \def\TrackedIsoCodeFromLanguage#1#2{%
407   \@tracklang@nameuse{@tracklang@#1@isofromlang@#2}%
408 }

```

dLanguageFromDialect

`\TrackedLanguageFromDialect{<dialect>}`

Fetches the language name from the given dialect.

```

409 \def\TrackedLanguageFromDialect#1{%
410   \@tracklang@nameuse{@tracklang@fromdialect@#1}%
411 }

```

DialectsFromLanguage

`\TrackedDialectsFromLanguage{<root language label>}`

Fetches the tracked dialects whose language is given by <root language label>.

```

412 \def\TrackedDialectsFromLanguage#1{%
413   \@tracklang@nameuse{@tracklang@todialect@#1}%
414 }

```

etterIsoCountryCode

```
415 \def\TwoLetterIsoCountryCode{3166-1}
```

etterIsoLanguageCode

```
416 \def\TwoLetterIsoLanguageCode{639-1}
```

etterIsoLanguageCode

```
417 \def\ThreeLetterIsoLanguageCode{639-2}
```

The ISO 639-1 and 639-2 codes are used to map the root language name to the ISO language code. The 3166-1 codes are used to map the dialect language to the ISO country code. The country code is omitted if ambiguous (for example, the language is spoken in multiple countries). Some “dialects” are just synonyms for a language name, such as “francais” or “frenchb”. Some of the languages have two ISO 639-2 codes designated as “B” (bibliographic) or “T” (terminology). In these cases the terminology code is used as the primary 639-2 code.

ackPredefinedDialect

```
\TrackPredefinedDialect{<dialect label>}
```

```
418 \def\TrackPredefinedDialect#1{%
419   \@tracklang@ifundef{@tracklang@add@#1}%
420   {%
421     \@tracklang@err{Dialect ‘#1’ is not predefined}{}%
422   }%
423   {\@tracklang@nameuse{@tracklang@add@#1}}%
424 }
```

acklang@add@acadian

```
425 \def\@tracklang@add@acadian{%
426   \AddTrackedDialect{acadian}{french}%
427   \AddTrackedIsoLanguage{639-1}{fr}{french}%
428 }
```

cklang@add@afrikaans

```
429 \def\@tracklang@add@afrikaans{%
430   \AddTrackedLanguage{afrikaans}%
431   \AddTrackedIsoLanguage{639-1}{af}{afrikaans}%
432   \AddTrackedIsoLanguage{639-2}{afr}{afrikaans}%
433 }
```

cklang@add@albanian

```
434 \def\@tracklang@add@albanian{%
435   \AddTrackedLanguage{albanian}%
436   \AddTrackedIsoLanguage{639-1}{sq}{albanian}%
437   \AddTrackedIsoLanguage{639-2-B}{alb}{albanian}%

```

```

438 \AddTrackedIsoLanguage{639-2-T}{sqi}{albanian}%
439 \AddTrackedIsoLanguage{639-2}{sqi}{albanian}%
440 }

acklang@add@amharic

441 \def\@tracklang@add@amharic{%
442 \AddTrackedLanguage{amharic}%
443 \AddTrackedIsoLanguage{639-1}{am}{amharic}%
444 \AddTrackedIsoLanguage{639-2}{amh}{amharic}%
445 }

lang@add@anglosaxon

446 \def\@tracklang@add@anglosaxon{%
447 \AddTrackedLanguage{anglosaxon}%
448 \AddTrackedIsoLanguage{639-2}{ang}{anglosaxon}%
449 }

racklang@add@apache

450 \def\@tracklang@add@apache{%
451 \AddTrackedLanguage{apache}%
452 \AddTrackedIsoLanguage{639-2}{apa}{apache}%
453 }

racklang@add@arabic

454 \def\@tracklang@add@arabic{%
455 \AddTrackedLanguage{arabic}%
456 \AddTrackedIsoLanguage{639-1}{ar}{arabic}%
457 \AddTrackedIsoLanguage{639-2}{ara}{arabic}%
458 }

cklang@add@armenian

459 \def\@tracklang@add@armenian{%
460 \AddTrackedLanguage{armenian}%
461 \AddTrackedIsoLanguage{639-1}{hy}{armenian}%
462 \AddTrackedIsoLanguage{639-2-B}{arm}{armenian}%
463 \AddTrackedIsoLanguage{639-2-T}{hye}{armenian}%
464 \AddTrackedIsoLanguage{639-2}{hye}{armenian}%
465 }

cklang@add@asturian

466 \def\@tracklang@add@asturian{%
467 \AddTrackedLanguage{asturian}%
468 \AddTrackedIsoLanguage{639-2}{ast}{asturian}%
469 }

cklang@add@american

470 \def\@tracklang@add@american{%
471 \AddTrackedDialect{american}{english}%

```



```

472 \AddTrackedIsoLanguage{639-1}{en}{english}%
473 \AddTrackedIsoLanguage{639-2}{eng}{english}%
474 \AddTrackedIsoLanguage{3166-1}{US}{american}%
475 }

```

lang@add@australian

```

476 \def\@tracklang@add@australian{%
477 \AddTrackedDialect{australian}{english}%
478 \AddTrackedIsoLanguage{639-1}{en}{english}%
479 \AddTrackedIsoLanguage{639-2}{eng}{english}%
480 \AddTrackedIsoLanguage{3166-1}{AU}{australian}%
481 }

```

cklang@add@austrian

```

482 \def\@tracklang@add@austrian{%
483 \AddTrackedDialect{austrian}{german}%
484 \AddTrackedIsoLanguage{639-1}{de}{german}%
485 \AddTrackedIsoLanguage{639-2-B}{ger}{german}%
486 \AddTrackedIsoLanguage{639-2-T}{deu}{german}%
487 \AddTrackedIsoLanguage{639-2}{deu}{german}%
488 \AddTrackedIsoLanguage{3166-1}{AT}{austrian}%
489 }

```

klang@add@naustrian

```

490 \def\@tracklang@add@naustrian{%
491 \AddTrackedDialect{naustrian}{german}%
492 \AddTrackedIsoLanguage{639-1}{de}{german}%
493 \AddTrackedIsoLanguage{639-2-B}{ger}{german}%
494 \AddTrackedIsoLanguage{639-2-T}{deu}{german}%
495 \AddTrackedIsoLanguage{639-2}{deu}{german}%
496 \AddTrackedIsoLanguage{3166-1}{AT}{naustrian}%
497 }

```

racklang@add@bahasa

```

498 \def\@tracklang@add@bahasa{%
499 \AddTrackedDialect{bahasa}{bahasai}%
500 \AddTrackedIsoLanguage{639-1}{id}{bahasai}%
501 \AddTrackedIsoLanguage{639-2}{ind}{bahasai}%
502 \AddTrackedIsoLanguage{3166-1}{IN}{bahasa}%
503 }

```

acklang@add@bahasai

```

504 \def\@tracklang@add@bahasai{%
505 \AddTrackedDialect{bahasai}{bahasai}%
506 \AddTrackedIsoLanguage{639-1}{id}{bahasai}%
507 \AddTrackedIsoLanguage{639-2}{ind}{bahasai}%
508 \AddTrackedIsoLanguage{3166-1}{IN}{bahasai}%
509 }

```

acklang@add@bahasam

```
510 \def\@tracklang@add@bahasam{%
511   \AddTrackedLanguage{bahasam}%
512   \AddTrackedIsoLanguage{639-1}{ms}{bahasam}%
513   \AddTrackedIsoLanguage{639-2-B}{may}{bahasam}%
514   \AddTrackedIsoLanguage{639-2-T}{msa}{bahasam}%
515   \AddTrackedIsoLanguage{639-2}{msa}{bahasam}%
516   \AddTrackedIsoLanguage{3166-1}{MY}{bahasam}%
517 }
```

acklang@add@basque

```
518 \def\@tracklang@add@basque{%
519   \AddTrackedLanguage{basque}%
520   \AddTrackedIsoLanguage{639-1}{eu}{basque}%
521   \AddTrackedIsoLanguage{639-2-B}{baq}{basque}%
522   \AddTrackedIsoLanguage{639-2-T}{eus}{basque}%
523   \AddTrackedIsoLanguage{639-2}{eus}{basque}%
524 }
```

acklang@add@bengali

```
525 \def\@tracklang@add@bengali{%
526   \AddTrackedLanguage{bengali}%
527   \AddTrackedIsoLanguage{639-1}{bn}{bengali}%
528   \AddTrackedIsoLanguage{639-2}{ben}{bengali}%
529 }
```

acklang@add@brazil

```
530 \def\@tracklang@add@brazil{%
531   \AddTrackedDialect{brazil}{portuges}%
532   \AddTrackedIsoLanguage{639-1}{pt}{portuges}%
533   \AddTrackedIsoLanguage{639-2}{por}{portuges}%
534   \AddTrackedIsoLanguage{3166-1}{BR}{brazil}%
535 }
```

acklang@add@brazilian

```
536 \def\@tracklang@add@brazilian{%
537   \AddTrackedDialect{brazilian}{portuges}%
538   \AddTrackedIsoLanguage{639-1}{pt}{portuges}%
539   \AddTrackedIsoLanguage{639-2}{por}{portuges}%
540   \AddTrackedIsoLanguage{3166-1}{BR}{brazilian}%
541 }
```

acklang@add@breton

```
542 \def\@tracklang@add@breton{%
543   \AddTrackedLanguage{breton}%
544   \AddTrackedIsoLanguage{639-1}{br}{breton}%
545   \AddTrackedIsoLanguage{639-2}{bre}{breton}%
546   \AddTrackedIsoLanguage{3166-1}{FR}{breton}%
547 }
```

acklang@add@british

```
548 \def\@tracklang@add@british{%  
549   \AddTrackedDialect{british}{english}%  
550   \AddTrackedIsoLanguage{639-1}{en}{english}%  
551   \AddTrackedIsoLanguage{639-2}{eng}{english}%  
552   \AddTrackedIsoLanguage{3166-1}{GB}{british}%  
553 }
```

cklang@add@bulgarian

```
554 \def\@tracklang@add@bulgarian{%  
555   \AddTrackedLanguage{bulgarian}%  
556   \AddTrackedIsoLanguage{639-1}{bg}{bulgarian}%  
557   \AddTrackedIsoLanguage{639-2}{bul}{bulgarian}%  
558 }
```

cklang@add@canadian

```
559 \def\@tracklang@add@canadian{%  
560   \AddTrackedDialect{canadian}{english}%  
561   \AddTrackedIsoLanguage{639-1}{en}{english}%  
562   \AddTrackedIsoLanguage{639-2}{eng}{english}%  
563   \AddTrackedIsoLanguage{3166-1}{CA}{canadian}%  
564 }
```

cklang@add@canadien

```
565 \def\@tracklang@add@canadien{%  
566   \AddTrackedDialect{canadien}{french}%  
567   \AddTrackedIsoLanguage{639-1}{fr}{french}%  
568   \AddTrackedIsoLanguage{639-2-B}{fre}{french}%  
569   \AddTrackedIsoLanguage{639-2-T}{fra}{french}%  
570   \AddTrackedIsoLanguage{639-2}{fra}{french}%  
571   \AddTrackedIsoLanguage{3166-1}{CA}{canadien}%  
572 }
```

acklang@add@catalan

```
573 \def\@tracklang@add@catalan{%  
574   \AddTrackedLanguage{catalan}%  
575   \AddTrackedIsoLanguage{639-1}{ca}{catalan}%  
576   \AddTrackedIsoLanguage{639-2}{cat}{catalan}%  
577 }
```

racklang@add@coptic

```
578 \def\@tracklang@add@coptic{%  
579   \AddTrackedLanguage{coptic}%  
580   \AddTrackedIsoLanguage{639-2}{cop}{coptic}%  
581 }
```

cklang@add@croatian

```
582 \def\@tracklang@add@croatian{%
```

```

583 \AddTrackedLanguage{croatian}%
584 \AddTrackedIsoLanguage{639-1}{hr}{croatian}%
585 \AddTrackedIsoLanguage{639-2}{hrv}{croatian}%
586 }

tracklang@add@czech
587 \def\@tracklang@add@czech{%
588 \AddTrackedLanguage{czech}%
589 \AddTrackedIsoLanguage{639-1}{cs}{czech}%
590 \AddTrackedIsoLanguage{639-2-B}{cze}{czech}%
591 \AddTrackedIsoLanguage{639-2-T}{ces}{czech}%
592 \AddTrackedIsoLanguage{639-2}{ces}{czech}%
593 }

tracklang@add@danish
594 \def\@tracklang@add@danish{%
595 \AddTrackedLanguage{danish}%
596 \AddTrackedIsoLanguage{639-1}{da}{danish}%
597 \AddTrackedIsoLanguage{639-2}{dan}{danish}%
598 }

tracklang@add@divehi
599 \def\@tracklang@add@divehi{%
600 \AddTrackedLanguage{divehi}%
601 \AddTrackedIsoLanguage{639-1}{dv}{divehi}%
602 \AddTrackedIsoLanguage{639-2}{div}{divehi}%
603 }

tracklang@add@dutch
604 \def\@tracklang@add@dutch{%
605 \AddTrackedLanguage{dutch}%
606 \AddTrackedIsoLanguage{639-1}{nl}{dutch}%
607 \AddTrackedIsoLanguage{639-2-B}{dut}{dutch}%
608 \AddTrackedIsoLanguage{639-2-T}{nld}{dutch}%
609 \AddTrackedIsoLanguage{639-2}{nld}{dutch}%
610 }

tracklang@add@english
611 \def\@tracklang@add@english{%
612 \AddTrackedLanguage{english}%
613 \AddTrackedIsoLanguage{639-1}{en}{english}%
614 \AddTrackedIsoLanguage{639-2}{eng}{english}%
615 }

tracklang@add@esperanto
616 \def\@tracklang@add@esperanto{%
617 \AddTrackedLanguage{esperanto}%
618 \AddTrackedIsoLanguage{639-1}{eo}{esperanto}%
619 \AddTrackedIsoLanguage{639-2}{epo}{esperanto}%
620 }

```

cklang@add@estonian

```
621 \def\@tracklang@add@estonian{%  
622   \AddTrackedLanguage{estonian}%  
623   \AddTrackedIsoLanguage{639-1}{et}{estonian}%  
624   \AddTrackedIsoLanguage{639-2}{est}{estonian}%  
625 }
```

tracklang@add@farsi

```
626 \def\@tracklang@add@farsi{%  
627   \AddTrackedLanguage{farsi}%  
628   \AddTrackedIsoLanguage{639-1}{fa}{farsi}%  
629   \AddTrackedIsoLanguage{639-2-B}{per}{farsi}%  
630   \AddTrackedIsoLanguage{639-2-T}{fas}{farsi}%  
631   \AddTrackedIsoLanguage{639-2}{fas}{farsi}%  
632 }
```

acklang@add@persian

```
633 \def\@tracklang@add@persian{%  
634   \AddTrackedDialect{persian}{farsi}%  
635   \AddTrackedIsoLanguage{639-1}{fa}{farsi}%  
636   \AddTrackedIsoLanguage{639-2-B}{per}{farsi}%  
637   \AddTrackedIsoLanguage{639-2-T}{fas}{farsi}%  
638   \AddTrackedIsoLanguage{639-2}{fas}{farsi}%  
639 }
```

acklang@add@finnish

```
640 \def\@tracklang@add@finnish{%  
641   \AddTrackedLanguage{finnish}%  
642   \AddTrackedIsoLanguage{639-1}{fi}{finnish}%  
643   \AddTrackedIsoLanguage{639-2}{fin}{finnish}%  
644 }
```

acklang@add@flemish

```
645 \def\@tracklang@add@flemish{%  
646   \AddTrackedDialect{flemish}{dutch}%  
647   \AddTrackedIsoLanguage{639-1}{nl}{dutch}%  
648   \AddTrackedIsoLanguage{639-2-B}{dut}{dutch}%  
649   \AddTrackedIsoLanguage{639-2-T}{nld}{dutch}%  
650   \AddTrackedIsoLanguage{639-2}{nld}{dutch}%  
651   \AddTrackedIsoLanguage{3166-1}{BE}{flemish}%  
652 }
```

cklang@add@français

```
653 \def\@tracklang@add@français{%  
654   \AddTrackedDialect{français}{french}%  
655   \AddTrackedIsoLanguage{639-1}{fr}{french}%  
656   \AddTrackedIsoLanguage{639-2-B}{fre}{french}%  
657   \AddTrackedIsoLanguage{639-2-T}{fra}{french}%  
658 }
```

```

658 \AddTrackedIsoLanguage{639-2}{fra}{french}%
659 }

acklang@add@frenchb

660 \def\@tracklang@add@frenchb{%
661 \AddTrackedDialect{frenchb}{french}%
662 \AddTrackedIsoLanguage{639-1}{fr}{french}%
663 \AddTrackedIsoLanguage{639-2-B}{fre}{french}%
664 \AddTrackedIsoLanguage{639-2-T}{fra}{french}%
665 \AddTrackedIsoLanguage{639-2}{fra}{french}%
666 }

acklang@add@french

667 \def\@tracklang@add@french{%
668 \AddTrackedLanguage{french}%
669 \AddTrackedIsoLanguage{639-1}{fr}{french}%
670 \AddTrackedIsoLanguage{639-2-B}{fre}{french}%
671 \AddTrackedIsoLanguage{639-2-T}{fra}{french}%
672 \AddTrackedIsoLanguage{639-2}{fra}{french}%
673 }

acklang@add@belgique

674 \def\@tracklang@add@belgique{%
675 \AddTrackedDialect{belgique}{french}%
676 \AddTrackedIsoLanguage{639-1}{fr}{french}%
677 \AddTrackedIsoLanguage{639-2-B}{fre}{french}%
678 \AddTrackedIsoLanguage{639-2-T}{fra}{french}%
679 \AddTrackedIsoLanguage{639-2}{fra}{french}%
680 \AddTrackedIsoLanguage{3166-1}{BE}{belgique}%
681 }

acklang@add@friulan

682 \def\@tracklang@add@friulan{%
683 \AddTrackedLanguage{friulan}%
684 \AddTrackedIsoLanguage{639-2}{fur}{friulan}%
685 \AddTrackedIsoLanguage{3166-1}{IT}{friulan}%
686 }

acklang@add@friulian

687 \def\@tracklang@add@friulian{%
688 \AddTrackedDialect{friulian}{friulan}%
689 \AddTrackedIsoLanguage{639-2}{fur}{friulan}%
690 \AddTrackedIsoLanguage{3166-1}{IT}{friulan}%
691 }

acklang@add@friulano

692 \def\@tracklang@add@friulano{%
693 \AddTrackedDialect{friulano}{friulan}%

```

```

694 \AddTrackedIsoLanguage{639-2}{fur}{friulan}%
695 \AddTrackedIsoLanguage{3166-1}{IT}{friulan}%
696 }

cklang@add@galician

697 \def\@tracklang@add@galician{%
698 \AddTrackedLanguage{galician}%
699 \AddTrackedIsoLanguage{639-1}{gl}{galician}%
700 \AddTrackedIsoLanguage{639-2}{glg}{galician}%
701 }

cklang@add@galicien

702 \def\@tracklang@add@galicien{%
703 \AddTrackedDialect{galicien}{galician}%
704 \AddTrackedIsoLanguage{639-1}{gl}{galician}%
705 \AddTrackedIsoLanguage{639-2}{glg}{galician}%
706 }

racklang@add@german

707 \def\@tracklang@add@german{%
708 \AddTrackedLanguage{german}%
709 \AddTrackedIsoLanguage{639-1}{de}{german}%
710 \AddTrackedIsoLanguage{639-2-B}{ger}{german}%
711 \AddTrackedIsoLanguage{639-2-T}{deu}{german}%
712 \AddTrackedIsoLanguage{639-2}{deu}{german}%
713 }

acklang@add@deutsch

714 \def\@tracklang@add@deutsch{%
715 \AddTrackedDialect{deutsch}{german}%
716 \AddTrackedIsoLanguage{639-1}{de}{german}%
717 \AddTrackedIsoLanguage{639-2-B}{ger}{german}%
718 \AddTrackedIsoLanguage{639-2-T}{deu}{german}%
719 \AddTrackedIsoLanguage{639-2}{deu}{german}%
720 }

acklang@add@ngerman

721 \def\@tracklang@add@ngerman{%
722 \AddTrackedDialect{ngerman}{german}%
723 \AddTrackedIsoLanguage{639-1}{de}{german}%
724 \AddTrackedIsoLanguage{639-2-B}{ger}{german}%
725 \AddTrackedIsoLanguage{639-2-T}{deu}{german}%
726 \AddTrackedIsoLanguage{639-2}{deu}{german}%
727 }

tracklang@add@greek

728 \def\@tracklang@add@greek{%
729 \AddTrackedLanguage{greek}%

```

```

730 \AddTrackedIsoLanguage{639-1}{el}{greek}%
731 \AddTrackedIsoLanguage{639-2-B}{gre}{greek}%
732 \AddTrackedIsoLanguage{639-2-T}{ell}{greek}%
733 \AddTrackedIsoLanguage{639-2}{ell}{greek}%
734 }

```

tracklang@add@hebrew

```

735 \def\@tracklang@add@hebrew{%
736 \AddTrackedLanguage{hebrew}%
737 \AddTrackedIsoLanguage{639-1}{he}{hebrew}%
738 \AddTrackedIsoLanguage{639-2}{heb}{hebrew}%
739 }

```

tracklang@add@hindi

```

740 \def\@tracklang@add@hindi{%
741 \AddTrackedLanguage{hindi}%
742 \AddTrackedIsoLanguage{639-1}{hi}{hindi}%
743 \AddTrackedIsoLanguage{639-2}{hin}{hindi}%
744 }

```

tracklang@add@hungarian

```

745 \def\@tracklang@add@hungarian{%
746 \AddTrackedDialect{hungarian}{magyar}%
747 \AddTrackedIsoLanguage{639-1}{hu}{magyar}%
748 \AddTrackedIsoLanguage{639-2}{hun}{magyar}%
749 \AddTrackedIsoLanguage{3166-1}{HU}{hungarian}%
750 }

```

tracklang@add@magyar

```

751 \def\@tracklang@add@magyar{%
752 \AddTrackedLanguage{magyar}%
753 \AddTrackedIsoLanguage{639-1}{hu}{magyar}%
754 \AddTrackedIsoLanguage{639-2}{hun}{magyar}%
755 }

```

tracklang@add@icelandic

```

756 \def\@tracklang@add@icelandic{%
757 \AddTrackedLanguage{icelandic}%
758 \AddTrackedIsoLanguage{639-1}{is}{icelandic}%
759 \AddTrackedIsoLanguage{639-2-B}{ice}{icelandic}%
760 \AddTrackedIsoLanguage{639-2-T}{isl}{icelandic}%
761 \AddTrackedIsoLanguage{639-2}{isl}{icelandic}%
762 }

```

tracklang@add@indon

```

763 \def\@tracklang@add@indon{%
764 \AddTrackedDialect{indon}{bahasai}%
765 \AddTrackedIsoLanguage{639-1}{id}{bahasai}%

```



```

766 \AddTrackedIsoLanguage{639-2}{ind}{bahasai}%
767 \AddTrackedIsoLanguage{3166-1}{IN}{indon}%
768 }

```

lang@add@indonesian

```

769 \def\@tracklang@add@indonesian{%
770 \AddTrackedDialect{indonesian}{bahasai}%
771 \AddTrackedIsoLanguage{639-1}{id}{bahasai}%
772 \AddTrackedIsoLanguage{639-2}{ind}{bahasai}%
773 \AddTrackedIsoLanguage{3166-1}{IN}{indonesian}%
774 }

```

ang@add@interlingua

```

775 \def\@tracklang@add@interlingua{%
776 \AddTrackedLanguage{interlingua}%
777 \AddTrackedIsoLanguage{639-1}{ia}{interlingua}%
778 \AddTrackedIsoLanguage{639-2}{ina}{interlingua}%
779 }

```

tracklang@add@irish

```

780 \def\@tracklang@add@irish{%
781 \AddTrackedLanguage{irish}%
782 \AddTrackedIsoLanguage{639-1}{ga}{irish}%
783 \AddTrackedIsoLanguage{639-2}{gle}{irish}%
784 }

```

acklang@add@gaeilge

```

785 \def\@tracklang@add@gaeilge{%
786 \AddTrackedDialect{gaeilge}{irish}%
787 \AddTrackedIsoLanguage{639-1}{ga}{irish}%
788 \AddTrackedIsoLanguage{639-2}{gle}{irish}%
789 }

```

acklang@add@italian

```

790 \def\@tracklang@add@italian{%
791 \AddTrackedLanguage{italian}%
792 \AddTrackedIsoLanguage{639-1}{it}{italian}%
793 \AddTrackedIsoLanguage{639-2}{ita}{italian}%
794 }

```

acklang@add@kannada

```

795 \def\@tracklang@add@kannada{%
796 \AddTrackedLanguage{kannada}%
797 \AddTrackedIsoLanguage{639-1}{kn}{kannada}%
798 \AddTrackedIsoLanguage{639-2}{kan}{kannada}%
799 }

```

\@tracklang@add@lao

```
800 \def\@tracklang@add@lao{%  
801   \AddTrackedLanguage{lao}%  
802   \AddTrackedIsoLanguage{639-1}{lo}{lao}%  
803   \AddTrackedIsoLanguage{639-2}{lao}{lao}%  
804 }
```

tracklang@add@latin

```
805 \def\@tracklang@add@latin{%  
806   \AddTrackedLanguage{latin}%  
807   \AddTrackedIsoLanguage{639-1}{la}{latin}%  
808   \AddTrackedIsoLanguage{639-2}{lat}{latin}%  
809 }
```

tracklang@add@latein

```
810 \def\@tracklang@add@latein{%  
811   \AddTrackedDialect{latein}{latin}%  
812   \AddTrackedIsoLanguage{639-1}{la}{latin}%  
813   \AddTrackedIsoLanguage{639-2}{lat}{latin}%  
814 }
```

acklang@add@latvian

```
815 \def\@tracklang@add@latvian{%  
816   \AddTrackedLanguage{latvian}%  
817   \AddTrackedIsoLanguage{639-1}{lv}{latvian}%  
818   \AddTrackedIsoLanguage{639-2}{lav}{latvian}%  
819 }
```

lang@add@lithuanian

```
820 \def\@tracklang@add@lithuanian{%  
821   \AddTrackedLanguage{lithuanian}%  
822   \AddTrackedIsoLanguage{639-1}{lt}{lithuanian}%  
823   \AddTrackedIsoLanguage{639-2}{lit}{lithuanian}%  
824 }
```

cklang@add@lsorbian

```
825 \def\@tracklang@add@lsorbian{%  
826   \AddTrackedLanguage{lsorbian}%  
827   \AddTrackedIsoLanguage{639-2}{dsb}{lsorbian}%  
828   \AddTrackedIsoLanguage{3166-1}{DE}{lsorbian}%  
829 }
```

ng@add@lowersorbian

```
830 \def\@tracklang@add@lowersorbian{%  
831   \AddTrackedDialect{lowersorbian}{lsorbian}%  
832   \AddTrackedIsoLanguage{639-2}{dsb}{lsorbian}%  
833   \AddTrackedIsoLanguage{3166-1}{DE}{lowersorbian}%  
834 }
```

tracklang@add@malay

```
835 \def\@tracklang@add@malay{%
836   \AddTrackedDialect{malay}{bahasam}%
837   \AddTrackedIsoLanguage{639-1}{ms}{bahasam}%
838   \AddTrackedIsoLanguage{639-2-B}{may}{bahasam}%
839   \AddTrackedIsoLanguage{639-2-T}{msa}{bahasam}%
840   \AddTrackedIsoLanguage{639-2}{msa}{bahasam}%
841   \AddTrackedIsoLanguage{3166-1}{MY}{malay}%
842 }
```

tracklang@add@meyalu

```
843 \def\@tracklang@add@meyalu{%
844   \AddTrackedDialect{meyalu}{bahasam}%
845   \AddTrackedIsoLanguage{639-1}{ms}{bahasam}%
846   \AddTrackedIsoLanguage{639-2-B}{may}{bahasam}%
847   \AddTrackedIsoLanguage{639-2-T}{msa}{bahasam}%
848   \AddTrackedIsoLanguage{639-2}{msa}{bahasam}%
849   \AddTrackedIsoLanguage{3166-1}{MY}{meyalu}%
850 }
```

tracklang@add@malayalam

```
851 \def\@tracklang@add@malayalam{%
852   \AddTrackedLanguage{malayalam}%
853   \AddTrackedIsoLanguage{639-1}{ml}{malayalam}%
854   \AddTrackedIsoLanguage{639-2}{mal}{malayalam}%
855 }
```

tracklang@add@marathi

```
856 \def\@tracklang@add@marathi{%
857   \AddTrackedLanguage{marathi}%
858   \AddTrackedIsoLanguage{639-1}{mr}{marathi}%
859   \AddTrackedIsoLanguage{639-2}{mar}{marathi}%
860 }
```

tracklang@add@newzealand

```
861 \def\@tracklang@add@newzealand{%
862   \AddTrackedDialect{newzealand}{english}%
863   \AddTrackedIsoLanguage{639-1}{en}{english}%
864   \AddTrackedIsoLanguage{639-2}{eng}{english}%
865   \AddTrackedIsoLanguage{3166-1}{NZ}{newzealand}%
866 }
```

\@tracklang@add@nko

```
867 \def\@tracklang@add@nko{%
868   \AddTrackedLanguage{nko}%
869   \AddTrackedIsoLanguage{639-2}{nqo}{nko}%
870 }
```

acklang@add@nynorsk

```
871 \def\@tracklang@add@nynorsk{%  
872   \AddTrackedDialect{nynorsk}{norsk}%  
873   \AddTrackedIsoLanguage{639-1}{nn}{norsk}%  
874   \AddTrackedIsoLanguage{639-2}{nno}{norsk}%  
875   \AddTrackedIsoLanguage{3166-1}{NO}{nynorsk}%  
876 }
```

cklang@add@norwegian

```
877 \def\@tracklang@add@norwegian{%  
878   \AddTrackedDialect{norwegian}{norsk}%  
879   \AddTrackedIsoLanguage{639-1}{nn}{norsk}%  
880   \AddTrackedIsoLanguage{639-2}{nno}{norsk}%  
881   \AddTrackedIsoLanguage{3166-1}{NO}{norwegian}%  
882 }
```

tracklang@add@norsk

```
883 \def\@tracklang@add@norsk{%  
884   \AddTrackedLanguage{norsk}%  
885   \AddTrackedIsoLanguage{639-1}{nn}{norsk}%  
886   \AddTrackedIsoLanguage{639-2}{nno}{norsk}%  
887   \AddTrackedIsoLanguage{3166-1}{NO}{norsk}%  
888 }
```

acklang@add@occitan

```
889 \def\@tracklang@add@occitan{%  
890   \AddTrackedLanguage{occitan}%  
891   \AddTrackedIsoLanguage{639-1}{oc}{occitan}%  
892   \AddTrackedIsoLanguage{639-2}{oci}{occitan}%  
893 }
```

ang@add@piedmontese

```
894 \def\@tracklang@add@piedmontese{%  
895   \AddTrackedLanguage{piedmontese}%  
896   \AddTrackedIsoLanguage{3166-1}{IT}{piedmontese}%  
897 }
```

lang@add@piemonteis

```
898 \def\@tracklang@add@piemonteis{%  
899   \AddTrackedDialect{piemonteis}{piedmontese}%  
900   \AddTrackedIsoLanguage{3166-1}{IT}{piemonteis}%  
901 }
```

racklang@add@polish

```
902 \def\@tracklang@add@polish{%  
903   \AddTrackedLanguage{polish}%  
904   \AddTrackedIsoLanguage{639-1}{pl}{polish}%  
905   \AddTrackedIsoLanguage{639-2}{pol}{polish}%  
906 }
```

add@polutonikogreek

```
907 \def\@tracklang@add@polutonikogreek{%
908   \AddTrackedDialect{polutonikogreek}{greek}%
909   \AddTrackedIsoLanguage{639-1}{el}{greek}%
910   \AddTrackedIsoLanguage{639-2-B}{gre}{greek}%
911   \AddTrackedIsoLanguage{639-2-T}{ell}{greek}%
912   \AddTrackedIsoLanguage{639-2}{ell}{greek}%
913 }
```

lang@add@polutoniko

```
914 \def\@tracklang@add@polutoniko{%
915   \AddTrackedDialect{polutoniko}{greek}%
916   \AddTrackedIsoLanguage{639-1}{el}{greek}%
917   \AddTrackedIsoLanguage{639-2-B}{gre}{greek}%
918   \AddTrackedIsoLanguage{639-2-T}{ell}{greek}%
919   \AddTrackedIsoLanguage{639-2}{ell}{greek}%
920 }
```

lang@add@portuguese

```
921 \def\@tracklang@add@portuguese{%
922   \AddTrackedDialect{portuguese}{portuges}%
923   \AddTrackedIsoLanguage{639-1}{pt}{portuges}%
924   \AddTrackedIsoLanguage{639-2}{por}{portuges}%
925 }
```

cklang@add@portuges

```
926 \def\@tracklang@add@portuges{%
927   \AddTrackedLanguage{portuges}%
928   \AddTrackedIsoLanguage{639-1}{pt}{portuges}%
929   \AddTrackedIsoLanguage{639-2}{por}{portuges}%
930 }
```

cklang@add@romanian

```
931 \def\@tracklang@add@romanian{%
932   \AddTrackedLanguage{romanian}%
933   \AddTrackedIsoLanguage{639-1}{ro}{romanian}%
934   \AddTrackedIsoLanguage{639-2-B}{rum}{romanian}%
935   \AddTrackedIsoLanguage{639-2-T}{ron}{romanian}%
936   \AddTrackedIsoLanguage{639-2}{ron}{romanian}%
937 }
```

acklang@add@romansh

```
938 \def\@tracklang@add@romansh{%
939   \AddTrackedLanguage{romansh}%
940   \AddTrackedIsoLanguage{639-1}{rm}{romansh}%
941   \AddTrackedIsoLanguage{639-2}{roh}{romansh}%
942 }
```

cklang@add@romansch

```
943 \def\@tracklang@add@romansch{%  
944   \AddTrackedDialect{romansch}{romansh}%  
945   \AddTrackedIsoLanguage{639-1}{rm}{romansh}%  
946   \AddTrackedIsoLanguage{639-2}{roh}{romansh}%  
947 }
```

cklang@add@rumantsch

```
948 \def\@tracklang@add@rumantsch{%  
949   \AddTrackedDialect{rumantsch}{romansh}%  
950   \AddTrackedIsoLanguage{639-1}{rm}{romansh}%  
951   \AddTrackedIsoLanguage{639-2}{roh}{romansh}%  
952 }
```

cklang@add@romanche

```
953 \def\@tracklang@add@romanche{%  
954   \AddTrackedDialect{romanche}{romansh}%  
955   \AddTrackedIsoLanguage{639-1}{rm}{romansh}%  
956   \AddTrackedIsoLanguage{639-2}{roh}{romansh}%  
957 }
```

cklang@add@russian

```
958 \def\@tracklang@add@russian{%  
959   \AddTrackedLanguage{russian}%  
960   \AddTrackedIsoLanguage{639-1}{ru}{russian}%  
961   \AddTrackedIsoLanguage{639-2}{rus}{russian}%  
962 }
```

cklang@add@russianb

```
963 \def\@tracklang@add@russianb{%  
964   \AddTrackedDialect{russianb}{russian}%  
965   \AddTrackedIsoLanguage{639-1}{ru}{russian}%  
966   \AddTrackedIsoLanguage{639-2}{rus}{russian}%  
967 }
```

cklang@add@samin

```
968 \def\@tracklang@add@samin{%  
969   \AddTrackedLanguage{samin}%  
970   \AddTrackedIsoLanguage{639-1}{se}{samin}%  
971   \AddTrackedIsoLanguage{639-2}{sme}{samin}%  
972 }
```

cklang@add@sanskrit

```
973 \def\@tracklang@add@sanskrit{%  
974   \AddTrackedLanguage{sanskrit}%  
975   \AddTrackedIsoLanguage{639-1}{sa}{sanskrit}%  
976   \AddTrackedIsoLanguage{639-2}{san}{sanskrit}%  
977 }
```

cklang@add@scottish

```
978 \def\@tracklang@add@scottish{%  
979   \AddTrackedLanguage{scottish}%  
980   \AddTrackedIsoLanguage{639-1}{gd}{scottish}%  
981   \AddTrackedIsoLanguage{639-2}{gla}{scottish}%  
982 }
```

racklang@add@gaelic

```
983 \def\@tracklang@add@gaelic{%  
984   \AddTrackedDialect{gaelic}{scottish}%  
985   \AddTrackedIsoLanguage{639-1}{gd}{scottish}%  
986   \AddTrackedIsoLanguage{639-2}{gla}{scottish}%  
987 }
```

acklang@add@serbian

```
988 \def\@tracklang@add@serbian{%  
989   \AddTrackedLanguage{serbian}%  
990   \AddTrackedIsoLanguage{639-1}{sr}{serbian}%  
991   \AddTrackedIsoLanguage{639-2}{srp}{serbian}%  
992 }
```

racklang@add@slovak

```
993 \def\@tracklang@add@slovak{%  
994   \AddTrackedLanguage{slovak}%  
995   \AddTrackedIsoLanguage{639-1}{sk}{slovak}%  
996   \AddTrackedIsoLanguage{639-2-B}{slo}{slovak}%  
997   \AddTrackedIsoLanguage{639-2-T}{slk}{slovak}%  
998   \AddTrackedIsoLanguage{639-2}{slk}{slovak}%  
999 }
```

acklang@add@slovene

```
1000 \def\@tracklang@add@slovene{%  
1001   \AddTrackedLanguage{slovene}%  
1002   \AddTrackedIsoLanguage{639-1}{sl}{slovene}%  
1003   \AddTrackedIsoLanguage{639-2}{slv}{slovene}%  
1004 }
```

cklang@add@slovenian

```
1005 \def\@tracklang@add@slovenian{%  
1006   \AddTrackedDialect{slovenian}{slovene}%  
1007   \AddTrackedIsoLanguage{639-1}{sl}{slovene}%  
1008   \AddTrackedIsoLanguage{639-2}{slv}{slovene}%  
1009 }
```

acklang@add@spanish

```
1010 \def\@tracklang@add@spanish{%  
1011   \AddTrackedLanguage{spanish}%  
1012   \AddTrackedIsoLanguage{639-1}{es}{spanish}%
```

```

1013 \AddTrackedIsoLanguage{639-2}{spa}{spanish}%
1014 }

tracklang@add@swedish
1015 \def\@tracklang@add@swedish{%
1016 \AddTrackedLanguage{swedish}%
1017 \AddTrackedIsoLanguage{639-1}{sv}{swedish}%
1018 \AddTrackedIsoLanguage{639-2}{swe}{swedish}%
1019 }

tracklang@add@syriac
1020 \def\@tracklang@add@syriac{%
1021 \AddTrackedLanguage{syriac}%
1022 \AddTrackedIsoLanguage{639-2}{syr}{syriac}%
1023 }

tracklang@add@tamil
1024 \def\@tracklang@add@tamil{%
1025 \AddTrackedLanguage{tamil}%
1026 \AddTrackedIsoLanguage{639-1}{ta}{tamil}%
1027 \AddTrackedIsoLanguage{639-2}{tam}{tamil}%
1028 }

tracklang@add@telugu
1029 \def\@tracklang@add@telugu{%
1030 \AddTrackedLanguage{telugu}%
1031 \AddTrackedIsoLanguage{639-1}{te}{telugu}%
1032 \AddTrackedIsoLanguage{639-2}{tel}{telugu}%
1033 }

@tracklang@add@thai
1034 \def\@tracklang@add@tai{%
1035 \AddTrackedLanguage{tai}%
1036 \AddTrackedIsoLanguage{639-2}{tai}{tai}%
1037 }

@tracklang@add@thai
1038 \def\@tracklang@add@thai{%
1039 \AddTrackedLanguage{thai}%
1040 \AddTrackedIsoLanguage{639-1}{th}{thai}%
1041 \AddTrackedIsoLanguage{639-2}{tha}{thai}%
1042 }

tracklang@add@tibetan
1043 \def\@tracklang@add@tibetan{%
1044 \AddTrackedLanguage{tibetan}%
1045 \AddTrackedIsoLanguage{639-1}{bo}{tibetan}%
1046 \AddTrackedIsoLanguage{639-2-B}{tib}{tibetan}%

```



```

1047 \AddTrackedIsoLanguage{639-2-T}{bod}{tibetan}%
1048 \AddTrackedIsoLanguage{639-2}{bod}{tibetan}%
1049 }

acklang@add@turkish

1050 \def\@tracklang@add@turkish{%
1051 \AddTrackedLanguage{turkish}%
1052 \AddTrackedIsoLanguage{639-1}{tr}{turkish}%
1053 \AddTrackedIsoLanguage{639-2}{tur}{turkish}%
1054 }

acklang@add@turkmen

1055 \def\@tracklang@add@turkmen{%
1056 \AddTrackedLanguage{turkmen}%
1057 \AddTrackedIsoLanguage{639-1}{tk}{turkmen}%
1058 \AddTrackedIsoLanguage{639-2}{tuk}{turkmen}%
1059 }

acklang@add@UKenglish

1060 \def\@tracklang@add@UKenglish{%
1061 \AddTrackedDialect{UKenglish}{english}%
1062 \AddTrackedIsoLanguage{639-1}{en}{english}%
1063 \AddTrackedIsoLanguage{639-2}{eng}{english}%
1064 \AddTrackedIsoLanguage{3166-1}{GB}{UKenglish}%
1065 }

acklang@add@ukrainian

1066 \def\@tracklang@add@ukrainian{%
1067 \AddTrackedLanguage{ukrainian}%
1068 \AddTrackedIsoLanguage{639-1}{uk}{ukrainian}%
1069 \AddTrackedIsoLanguage{639-2}{ukr}{ukrainian}%
1070 \AddTrackedIsoLanguage{3166-1}{UA}{ukrainian}%
1071 }

acklang@add@ukraineб

1072 \def\@tracklang@add@ukraineб{%
1073 \AddTrackedDialect{ukraineб}{ukrainian}%
1074 \AddTrackedIsoLanguage{639-1}{uk}{ukrainian}%
1075 \AddTrackedIsoLanguage{639-2}{ukr}{ukrainian}%
1076 \AddTrackedIsoLanguage{3166-1}{UA}{ukraineб}%
1077 }

acklang@add@ukraine

1078 \def\@tracklang@add@ukraine{%
1079 \AddTrackedDialect{ukraine}{ukrainian}%
1080 \AddTrackedIsoLanguage{639-1}{uk}{ukrainian}%
1081 \AddTrackedIsoLanguage{639-2}{ukr}{ukrainian}%
1082 \AddTrackedIsoLanguage{3166-1}{UA}{ukraine}%
1083 }

```

@tracklang@add@urdu

```
1084 \def\@tracklang@add@urdu{%
1085   \AddTrackedLanguage{urdu}%
1086   \AddTrackedIsoLanguage{639-1}{ur}{urdu}%
1087   \AddTrackedIsoLanguage{639-2}{urd}{urdu}%
1088 }
```

cklang@add@usorbian

```
1089 \def\@tracklang@add@usorbian{%
1090   \AddTrackedLanguage{usorbian}%
1091   \AddTrackedIsoLanguage{639-2}{dsb}{usorbian}%
1092   \AddTrackedIsoLanguage{3166-1}{DE}{usorbian}%
1093 }
```

ng@add@upporsorbian

```
1094 \def\@tracklang@add@upporsorbian{%
1095   \AddTrackedDialect{upporsorbian}{usorbian}%
1096   \AddTrackedIsoLanguage{639-2}{hsb}{usorbian}%
1097   \AddTrackedIsoLanguage{3166-1}{DE}{upporsorbian}%
1098 }
```

klang@add@USenglish

```
1099 \def\@tracklang@add@USenglish{%
1100   \AddTrackedDialect{USenglish}{english}%
1101   \AddTrackedIsoLanguage{639-1}{en}{english}%
1102   \AddTrackedIsoLanguage{639-2}{eng}{english}%
1103   \AddTrackedIsoLanguage{3166-1}{US}{USenglish}%
1104 }
```

klang@add@valencian

```
1105 \def\@tracklang@add@valencian{%
1106   \AddTrackedDialect{valencian}{catalan}%
1107   \AddTrackedIsoLanguage{639-1}{ca}{catalan}%
1108   \AddTrackedIsoLanguage{639-2}{cat}{catalan}%
1109 }
```

klang@add@valencien

```
1110 \def\@tracklang@add@valencien{%
1111   \AddTrackedDialect{valencien}{catalan}%
1112   \AddTrackedIsoLanguage{639-1}{ca}{catalan}%
1113   \AddTrackedIsoLanguage{639-2}{cat}{catalan}%
1114 }
```

lang@add@vietnamese

```
1115 \def\@tracklang@add@vietnamese{%
1116   \AddTrackedLanguage{vietnamese}%
1117   \AddTrackedIsoLanguage{639-1}{vi}{vietnamese}%
1118   \AddTrackedIsoLanguage{639-2}{vie}{vietnamese}%
1119 }
```

acklang@add@walloon

```
1120 \def\@tracklang@add@walloon{%
1121   \AddTrackedLanguage{walloon}%
1122   \AddTrackedIsoLanguage{639-1}{wa}{walloon}%
1123   \AddTrackedIsoLanguage{639-2}{wln}{walloon}%
1124 }
```

tracklang@add@welsh

```
1125 \def\@tracklang@add@welsh{%
1126   \AddTrackedLanguage{welsh}%
1127   \AddTrackedIsoLanguage{639-1}{cy}{welsh}%
1128   \AddTrackedIsoLanguage{639-2-B}{wel}{welsh}%
1129   \AddTrackedIsoLanguage{639-2-T}{cym}{welsh}%
1130   \AddTrackedIsoLanguage{639-2}{cym}{welsh}%
1131 }
```

acklang@add@cymraeg

```
1132 \def\@tracklang@add@cymraeg{%
1133   \AddTrackedDialect{cymraeg}{welsh}%
1134   \AddTrackedIsoLanguage{639-1}{cy}{welsh}%
1135   \AddTrackedIsoLanguage{639-2-B}{wel}{welsh}%
1136   \AddTrackedIsoLanguage{639-2-T}{cym}{welsh}%
1137   \AddTrackedIsoLanguage{639-2}{cym}{welsh}%
1138 }
```

\IfTrackedLanguage

<code>\IfTrackedLanguage{<language>}{<true part>}{<false part>}</code>
--

```
1139 \long\def\IfTrackedLanguage#1#2#3{%
1140   \end{macrocode}
1141 % First find out if the language name is empty.
1142 %   \begin{macrocode}
1143   \edef\@tracklang@element{#1}%
1144   \ifx\@tracklang@element\empty
1145     #3%
1146   \else
1147     \expandafter\@tracklang@ifinlist\expandafter{\@tracklang@element}%
1148     \@tracklang@languages
1149     {%
1150       #2%
1151     }%
1152     {%
1153       Not in list, so do false part.
1154       #3%
```

```

1154     }%
1155   \fi
1156 }

```

\IfTrackedDialect \IfTrackedDialect{<dialect>}{<true part>}{<false part>}

```

1157 \long\def\IfTrackedDialect#1#2#3{%
1158   \@tracklang@ifundef{@tracklang@fromdialect@#1}{#3}{#2}%
1159 }

```

\IfTrackedIsoCode \IfTrackedIsoCode{<code type>}{<code>}{<true part>}{<false part>}

```

1160 \long\def\IfTrackedIsoCode#1#2#3#4{%
1161   \@tracklang@ifundef{@tracklang@#1@isotolang@#2}{#4}{#3}%
1162 }

```

edLanguageHasIsoCode \IfTrackedLanguageHasIsoCode{<code type>}{<language>}{<true part>}{<false part>}

```

1163 \long\def\IfTrackedLanguageHasIsoCode#1#2#3#4{%
1164   \@tracklang@ifundef{@tracklang@#1@isofromlang@#2}{#4}{#3}%
1165 }

```

ForEachTrackedLanguage \ForEachTrackedLanguage{<cs>}{<body>}

Iterates through the list of tracked languages. On each iteration <cs> is set to the language tag and <body> is performed.

```

1166 \long\def\ForEachTrackedLanguage#1#2{%
1167   \@tracklang@for#1:=\@tracklang@languages\do{#2}%
1168 }

```

ForEachTrackedDialect \ForEachTrackedDialect{<cs>}{<body>}

Iterates through the list of tracked dialects. On each iteration <cs> is set to the dialect tag and <body> is performed.

```

1169 \long\def\ForEachTrackedDialect#1#2{%
1170   \@tracklang@for#1:=\@tracklang@dialects\do{#2}%
1171 }

```

AnyTrackedLanguages

```

1172 \long\def\AnyTrackedLanguages#1#2{%

```

```

1173 \ifx\@tracklang@languages\empty
1174   #2%
1175 \else
1176   #1%
1177 \fi
1178 }

```

edLanguageFileExists

`\IfTrackedLanguageFileExists{<dialect>}{<prefix>}{<suffix>}{<true part>}{<false part>}`

Determines if the file `<prefix><tag><suffix>` exists, where `<tag>` is an ISO code or ISO codes identifying the language. If `<dialect>` hasn't been identified as a tracked dialect, this just does `<false part>`, otherwise this first tries with `<tag>` set to `<dialect>`, then tries with `<tag>` set to the root language label for `<dialect>`, then tries with `<tag>` set to `<ISO 639-1 code>-<ISO 3166-1 code>`, then tries with `<tag>` set to `<ISO 639-1 code>`, then tries with `<tag>` set to `<ISO 639-2 code>-<ISO 3166-1 code>`, then tries with `<tag>` set to `<ISO 639-2 code>`. If the file `<prefix><tag><suffix>` exists, `\CurrentTrackedTag` is set to `<tag>` and `<true part>` is performed, otherwise `<false part>` is performed.

```

1179 \long\def\IfTrackedLanguageFileExists#1#2#3#4#5{%
1180   \def\CurrentTrackedTag{}%
1181   \def\CurrentTrackedDialect{#1}%
1182   \def\CurrentTrackedLanguage{}%
1183   \def\CurrentTrackedRegion{}%
1184   \def\CurrentTrackedIsoCode{}%
1185   \IfTrackedDialect{#1}%
1186   {%

```

Fetch the region if defined.

```

1187   \IfTrackedLanguageHasIsoCode{3166-1}{#1}%
1188   {%
1189     \edef\CurrentTrackedRegion{%
1190       \TrackedIsoCodeFromLanguage{3166-1}{#1}}%
1191   }%
1192   {%

```

Fetch the root language.

```

1193   \edef\CurrentTrackedLanguage{\TrackedLanguageFromDialect{#1}}%

```

Try just the dialect label.

```

1194   \edef\CurrentTrackedTag{#1}%
1195   \@tracklang@iffileexists{#2\CurrentTrackedTag#3}%
1196   {#4}%
1197   {%

```

No file found for dialect label, try `<ISO 639-1>-<ISO 3166-1>` next.

```

1198   \IfTrackedLanguageHasIsoCode
1199   {639-1}{\CurrentTrackedLanguage}

```

```

1200      {%
1201      \edef\CurrentTrackedIsoCode{%
1202      \TrackedIsoCodeFromLanguage
1203      {639-1}{\CurrentTrackedLanguage}}%
1204      \ifx\CurrentTrackedRegion\@empty

```

No region, just try ISO 639-1 code

```

1205      \let\CurrentTrackedTag\CurrentTrackedIsoCode
1206      \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1207      {#4}%
1208      {%

```

No region and file for just *(ISO 639-1 code)*. Try ISO 639-2 code.

```

1209      \IfTrackedLanguageHasIsoCode
1210      {639-2}{\CurrentTrackedLanguage}
1211      {%
1212      \edef\CurrentTrackedIsoCode{%
1213      \TrackedIsoCodeFromLanguage
1214      {639-2}{\CurrentTrackedLanguage}}%
1215      \let\CurrentTrackedTag\CurrentTrackedIsoCode
1216      \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1217      {#4}%
1218      {%

```

Try root language.

```

1219      \let\CurrentTrackedTag\CurrentTrackedLanguage
1220      \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}{#4}{#5}%
1221      }%
1222      }%
1223      {%

```

No region, no ISO 639-1 code and no ISO 639-2 code. No file found for dialect label, try root language label next.

```

1224      \let\CurrentTrackedTag\CurrentTrackedLanguage
1225      \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}{#4}{#5}%
1226      }%
1227      }%
1228      \else

```

Has a region so try *(ISO 639-1 code)*-*(region)*

```

1229      \edef\CurrentTrackedTag{%
1230      \CurrentTrackedIsoCode-\CurrentTrackedRegion}%
1231      \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1232      {#4}%
1233      {%

```

Has a region but *(ISO 639-1 code)*-*(region)* not found, so try *(ISO 639-2 code)*-*(region)*

```

1234      \IfTrackedLanguageHasIsoCode
1235      {639-2}{\CurrentTrackedLanguage}
1236      {%

```

```

1237         \let\org@currenttrackedisocode\CurrentTrackedIsoCode
1238         \edef\CurrentTrackedIsoCode{%
1239             \TrackedIsoCodeFromLanguage
1240             {639-2}\CurrentTrackedLanguage}}%
1241         \edef\CurrentTrackedTag{%
1242             \CurrentTrackedIsoCode-\CurrentTrackedRegion}%
1243         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1244         {#4}%
1245         {%

```

Has a region but *ISO 639-1 code* - *region* not found and *ISO 639-2 code* - *region* not found, so try just *ISO 639-1 code*

```

1246         \let\CurrentTrackedTag\org@currenttrackedisocode
1247         \let\org@currenttrackedisocode\CurrentTrackedIsoCode
1248         \let\CurrentTrackedIsoCode\CurrentTrackedTag
1249         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1250         {#4}%
1251         {%

```

Try just the ISO 639-2 code.

```

1252         \let\CurrentTrackedIsoCode\org@currenttrackedisocode
1253         \let\CurrentTrackedTag\CurrentTrackedIsoCode
1254         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1255         {#4}%
1256         {%

```

Try just the region.

```

1257         \let\CurrentTrackedTag\CurrentTrackedRegion
1258         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1259         {#4}%
1260         {%

```

Try the root language.

```

1261         \let\CurrentTrackedTag\CurrentTrackedLanguage
1262         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}{#4}{#5}%
1263         }%
1264     }%
1265 }%
1266 }%
1267 }%
1268 {%

```

Has a region but *ISO 639-1 code* - *region* not found, and no ISO 639-2 code so just try *ISO 639-1 code*

```

1269         \let\CurrentTrackedTag\CurrentTrackedIsoCode
1270         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1271         {#4}%
1272         {%

```

Just try the region

```

1273         \let\CurrentTrackedTag\CurrentTrackedRegion

```

```

1274         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1275         {#4}%
1276         {%

```

Try the root language.

```

1277         \let\CurrentTrackedTag\CurrentTrackedLanguage
1278         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}{#4}{#5}%
1279         }%
1280     }%
1281 }%
1282 }%
1283 \fi
1284 }%
1285 {%

```

No ISO 639-1 code. Try ISO 639-2 code.

```

1286     \IfTrackedLanguageHasIsoCode
1287     {639-2}{\CurrentTrackedLanguage}
1288     {%
1289         \edef\CurrentTrackedIsoCode{%
1290             \TrackedIsoCodeFromLanguage
1291             {639-2}{\CurrentTrackedLanguage}}%
1292     \ifx\CurrentTrackedRegion\@empty

```

Has ISO 639-2 code but no region

```

1293         \let\CurrentTrackedTag\CurrentTrackedIsoCode
1294         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1295         {#4}%
1296         {%

```

Try the root language.

```

1297         \let\CurrentTrackedTag\CurrentTrackedLanguage
1298         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}{#4}{#5}%
1299         }%
1300     \else

```

Has ISO 639-2 code and a region. Try *ISO 639-2*–*region* first.

```

1301         \edef\CurrentTrackedTag{%
1302             \CurrentTrackedIsoCode–\CurrentTrackedRegion}%
1303         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1304         {#4}%
1305         {%

```

Has ISO 639-2 code and a region but *ISO 639-2*–*region* doesn't exist. Try just *ISO 639-2*

```

1306         \let\CurrentTrackedTag\CurrentTrackedIsoCode
1307         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1308         {#4}%
1309         {%

```

Has ISO 639-2 code and a region but *ISO 639-2* doesn't exist. Try just *region*

```

1310         \let\CurrentTrackedTag\CurrentTrackedRegion

```



```

1311         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1312         {#4}%
1313         {%

```

Try the root language.

```

1314         \let\CurrentTrackedTag\CurrentTrackedLanguage
1315         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}{#4}{#5}%
1316     }%
1317 }%
1318 }%
1319 \fi
1320 }%
1321 {%

```

No ISO 639-1 code or ISO 639-2 code. Try just the region.

```

1322     \ifx\CurrentTrackedRegion\@empty

```

No region. Try the root language.

```

1323         \let\CurrentTrackedTag\CurrentTrackedLanguage
1324         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}{#4}{#5}%
1325     \else

```

Try the region.

```

1326         \let\CurrentTrackedTag\CurrentTrackedRegion
1327         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}%
1328         {#4}%
1329         {%

```

Try the root language.

```

1330         \let\CurrentTrackedTag\CurrentTrackedLanguage
1331         \@tracklang@ifFileExists{#2\CurrentTrackedTag#3}{#4}{#5}%
1332     }%
1333     \fi
1334 }%
1335 }%
1336 }%
1337 }%
1338 {#5}% unknown dialect
1339 }

```

Index

Symbols	
\@tracklang@ifFileExists	35
\@tracklang@add	36
\@tracklang@add@UKenglish ...	57
\@tracklang@add@USenglish ...	58
\@tracklang@add@acadian	39
\@tracklang@add@afrikaans ...	39
\@tracklang@add@albanian	39
\@tracklang@add@american	40
\@tracklang@add@amharic	40
\@tracklang@add@anglosaxon ..	40
\@tracklang@add@apache	40
\@tracklang@add@arabic	40
\@tracklang@add@armenian	40
\@tracklang@add@asturian	40
\@tracklang@add@australian ..	41
\@tracklang@add@austrian	41
\@tracklang@add@bahasa	41
\@tracklang@add@bahasai	41
\@tracklang@add@bahasam	42
\@tracklang@add@basque	42
\@tracklang@add@belgique	46
\@tracklang@add@bengali	42
\@tracklang@add@brazil	42
\@tracklang@add@brazilian ...	42
\@tracklang@add@breton	42
\@tracklang@add@british	43
\@tracklang@add@bulgarian ...	43
\@tracklang@add@canadian	43
\@tracklang@add@canadien	43
\@tracklang@add@catalan	43
\@tracklang@add@coptic	43
\@tracklang@add@croatian	43
\@tracklang@add@cymraeg	59
\@tracklang@add@czech	44
\@tracklang@add@danish	44
\@tracklang@add@deutsch	47
\@tracklang@add@divehi	44
\@tracklang@add@dutch	44
\@tracklang@add@english	44
\@tracklang@add@esperanto ...	44
\@tracklang@add@estonian	45
\@tracklang@add@farsi	45
\@tracklang@add@finnish	45
\@tracklang@add@flemish	45
\@tracklang@add@francais	45
\@tracklang@add@french	46
\@tracklang@add@frenchb	46
\@tracklang@add@friulan	46
\@tracklang@add@friulano	46
\@tracklang@add@friulian	46
\@tracklang@add@gaeilge	49
\@tracklang@add@gaelic	55
\@tracklang@add@galician	47
\@tracklang@add@galicien	47
\@tracklang@add@german	47
\@tracklang@add@greek	47
\@tracklang@add@hebrew	48
\@tracklang@add@hindi	48
\@tracklang@add@hungarian ...	48
\@tracklang@add@icelandic ...	48
\@tracklang@add@indon	48
\@tracklang@add@indonesian ..	49
\@tracklang@add@interlingua .	49
\@tracklang@add@irish	49
\@tracklang@add@italian	49
\@tracklang@add@kannada	49
\@tracklang@add@lao	50
\@tracklang@add@latein	50
\@tracklang@add@latin	50
\@tracklang@add@latvian	50
\@tracklang@add@lithuanian ..	50
\@tracklang@add@lowersorbian	50
\@tracklang@add@lsorbian	50
\@tracklang@add@magyar	48
\@tracklang@add@malay	51
\@tracklang@add@malayalam ...	51
\@tracklang@add@marathi	51
\@tracklang@add@meyalu	51
\@tracklang@add@naustrian ...	41
\@tracklang@add@newzealand ..	51
\@tracklang@add@ngerman	47
\@tracklang@add@nko	51
\@tracklang@add@norsk	52
\@tracklang@add@norwegian ...	52
\@tracklang@add@nynorsk	52
\@tracklang@add@occitan	52
\@tracklang@add@persian	45
\@tracklang@add@piedmontese .	52
\@tracklang@add@piemonteis ..	52
\@tracklang@add@polish	52
\@tracklang@add@polutoniko ..	53

\@tracklang@add@polutonikogreek 53
\@tracklang@add@portuges 53
\@tracklang@add@portuguese .. 53
\@tracklang@add@romanche 54
\@tracklang@add@romanian 53
\@tracklang@add@romansch 54
\@tracklang@add@romansh 53
\@tracklang@add@rumantsch ... 54
\@tracklang@add@russian 54
\@tracklang@add@russianb 54
\@tracklang@add@samin 54
\@tracklang@add@sanskrit 54
\@tracklang@add@scottish 55
\@tracklang@add@serbian 55
\@tracklang@add@slovak 55
\@tracklang@add@slovene 55
\@tracklang@add@slovenian ... 55
\@tracklang@add@spanish 55
\@tracklang@add@swedish 56
\@tracklang@add@syriac 56
\@tracklang@add@tamil 56
\@tracklang@add@telugu 56
\@tracklang@add@thai 56
\@tracklang@add@tibetan 56
\@tracklang@add@turkish 57
\@tracklang@add@turkmen 57
\@tracklang@add@ukraine 57
\@tracklang@add@ukraineb 57
\@tracklang@add@ukrainian ... 57
\@tracklang@add@upporsorbian 58
\@tracklang@add@urdu 58
\@tracklang@add@usorbian 58
\@tracklang@add@valencian ... 58
\@tracklang@add@valencien ... 58
\@tracklang@add@vietnamese .. 58
\@tracklang@add@walloon 59
\@tracklang@add@welsh 59
\@tracklang@dialects 36
\@tracklang@err 35
\@tracklang@for 34
\@tracklang@ifinlist 36
\@tracklang@ifundef 34
\@tracklang@languages 36
\@tracklang@namedef 35
\@tracklang@nameuse 35
\@tracklang@nnil 34

A

\AddTrackedDialect 25,37

\AddTrackedIsoLanguage ... 25,37
\AddTrackedLanguage 25,37
animals.sty package 10
\AnyTrackedLanguages 3,60

B

babel package
..... 1–3, 11, 14, 17, 19, 24, 31, 33

C

\CurrentTrackedDialect .. 8, 14, 15
\CurrentTrackedIsoCode 7–9
\CurrentTrackedLanguage 7, 9
\CurrentTrackedRegion 7–9
\CurrentTrackedTag 7, 8

F

\ForEachTrackedDialect .. 2, 3, 60
\ForEachTrackedLanguage ... 3, 60

I

\IfTrackedDialect 3, 60
\IfTrackedIsoCode 5, 60
\IfTrackedLanguage 3, 59
\IfTrackedLanguageFile 14
\IfTrackedLanguageFileExists
..... 7, 8, 14, 20, 61
\IfTrackedLanguageHasIsoCode
..... 4, 60

N

ngerman package 2, 3, 19, 31

P

polyglossia package
..... 1–3, 11, 14, 16, 18, 20, 31

T

\ThisDialect 9
\ThreeLetterIsoLanguageCode 5, 39
\TrackedDialectsFromLanguage
..... 4, 38
\TrackedIsoCodeFromLanguage 5, 38
\TrackedLanguageFromDialect 4, 38
\TrackedLanguageFromIsoCode 5, 38
\TrackPredefinedDialect .. 26, 39
translator package 2, 3, 19, 31, 32
\TwoLetterIsoCountryCode .. 5, 39
\TwoLetterIsoLanguageCode . 5, 39