The **philex** Package

version 1.3

Peter Pagin
peter.pagin@philosophy.su.se

February 27, 2015, 14:07
License type: lppl

Changes since version 1.2

– Added automatic handling of long environment names.
– Added draft mode, with margin printing of labels.
– Bugfix: Broken cross-references to \lbb environments with optional names, now fixed.

Manual

The **philex** package is intended for use in subjects like philosophy and linguistics, where there is a need both of many example sentences and of named principles, and of cross-references to them in the text. Sometimes one wants cross-reference to environment numbers (or prefix plus number), sometimes to a non-numbered named environment, and sometimes to the content of the environment, for repeating the content in running text or in another environment. The package is a small addition to Wolfgang Sternefeld’s **linguex** package and requires **linguex**. As of version 0.2 it also requires **calc**. **philex** will load **linguex** if it is installed. The purpose of **philex** is to add functions for more flexible cross-referencing, for repeating and embedding named or numbered environments, **hyperref** support and a some further formatting options.

As of version 0.5 **philex** writes cross-ref information to the .aux file and reads that information at the beginning of the subsequent run of typesetting. This allows forward cross-referencing to environments that are named or numbered+named, such as (PP) on p. 6,
in analogy to the `\label`-reference system of LaTeX itself. As of version 0.9 hyperlinks produced by `hyperref` are created also for named environments with the `<hyper>` package option.

1 Calling the package

`philex` is called by adding `\usepackage[<package options>]{philex}` in the preamble. This will also automatically load the `linguex` package, as well as `xspace` and `cgloss4e`, which are used by `linguex`. If `linguex` is called separately, it should be called before `philex`, since `philex` (marginally) redefines some commands in `linguex`. `philex` also calls the `calc`, `ifthen` and `suffix` packages.

2 Package options

`<hyper>` With the `<hyper>` option hyperlinks will be created by cross-references to named environments (such as `\lbp`). This requires that the `hyperref` package is called (should be called last). If `hyperref` is called without the `<hyper>` option, hyperlinks will anyway be created from the usual `philex` cross-references that make use of the basic label- and cross-reference function in LaTeX.

`<draft>` With the `<draft>` option, labels will be printed in boldface in the margins, top-level labels in the left-hand margin, and sub-level labels in the right-hand margin. This is intended as a help to writing cross-reference intensive texts. Labels are often hard to remember, and the environment to be cross-referenced is usually easier to find in the pdf output than in the source file. Finding the label printed there is then convenient. The vertical alignment of the margin-printed labels with the `philex` environment itself is not exact. Occasionally, the label does not follow the environment across a page break (I have managed to fix this only partially), but usually the alignment is quite good.

The horizontal position of labels printed in the left margin can be adjusted by means of `\renewcommand{\philmarginfactor}{}`. Default is 1.2. Higher values push the labels closer to the page edge.

`<oldpunct>` With the `<oldpunct>` package option the old punctuation commands (see below) will still work. This is for the purpose of using more recent versions of `philex` with older files, without needing search-and-replace.
3 Top-level environments

The basic environment command of philex is `\lb{}`. It takes two obligatory arguments. The first argument is the label of the environment, and the second is the content of the environment itself. `\lb` uses the `\ex` formatting of linguex, and thereby the ExNo counter. Type

`\lb{nice}{This is a nice day. \ Not too hot.}`

and the result will be

(1) This is a nice day.
    Not too hot.

in case it is the first `\lb`-environment in your document.

In linguex you leave a blank line after sentence to close the environment. The blank line gives a `\par`-command to TeX. In philex the `\par`-command is built into the top-level environments. So you leave a blank line after an `\lb`-environment only if you want to start a new paragraph. The reason for this change is that only this way will the philex package work together with the extract package by Hendri Adriaens, which can be used for extracting material, e.g. for generating a handout, from documents making use of philex. linguex comes together with Sternefeld’s linguho package, which has the function of generating a handout from documents using linguex. linguho does not work with philex, since the linguex environments are not explicit in the document. The corresponding function, only more flexible, can be served with the extract package.

Update with `\lbu` In many cases you will want to produce a variant of the original sentence, which will have the same name or number as the original, but with an added suffix, like a letter or a prime symbol. This is achieved with the `\lbu{}`-command. It takes four obligatory arguments. The first is the label of the new sentence, the second is the label of the environment you are giving a variant of, the third is the suffix you want to add, and the fourth is the new content itself.

`\lbu{nicy}{nice}{\('\)}{But that yesterday was even nicer.}`

produces

(1') But yesterday was even nicer.
You may also want to have named environments, such as in (W) below (exemplifying forward cross-reference)

(W) Life is wonderful!

These are produced with the \lbp{}{}{}-command which makes use of the optional argument to \ex{} in linguex. If you later change the name of the principle, or the formulation, the changes are reflected in the cross-references and repetitions after the next typesetting, as long as you do not change the labels.

And updates with \lbu, like

(W+) Life is terrific!

work as before. Concerning long names and horizontal alignment, see Section ??.

In case you wish a list of independently numbered principles with a shared name stem, there are four independent series. Use one of the \lbpx-commands, where \texttt{x} is one of a,b,c,d. You can format the listing with \bpxformat{}{}{}, where again \texttt{x} has the corresponding value. The first argument sets the numbering style (see below), the second the prefix and the third argument the suffix. Then the \lbpx-command picks it up:

\begin{verbatim}
\bpaformat{1}{(T}{)}
\lbpa{kno}{John is in the know.}
\lbpa{hu}{Elsa is, too.}
\lbpa{kne}{Alfred agrees.}
\end{verbatim}

produces

(T1) John is in the know.
(T2) Elsa is, too.
(T3) Alfred agrees.

These enumerations are independent of ExNO and of each other, and the counters (bpx) are reset with the relevant \bpxformat. They still work with sub-clauses, cross-references and the rest of the apparatus. Note that these enumerations do not have to be contiguous:

(T4) But he shouldn’t.
Simple counter

There is also an additional counter, bna, with associated commands: the \bns command resets the counter; the \bn command steps the counter, prints its current value, and produces a \quad space. Useful for listing a number of comments.

Centering

All the top-level environments take an optional argument [c] for centering the text.

\lb[c]{compo}{\[\mu(\sigma(t_{1},\ldots,t_{n})) = r_{\sigma}(\mu(t_{1}),\ldots,\mu(t_{n}))\]}

produces

\begin{equation}
\mu(\sigma(t_{1},\ldots,t_{n})) = r_{\sigma}(\mu(t_{1}),\ldots,\mu(t_{n}))
\end{equation}

In this example a displaymath environment (\[\] or $$..$$) has been used. This produces extra vertical space above and below, suitable for equations. With centering and in-line math environment (\(\ldots\) or $..$), no extra vertical space is produced.

The default for philex is centering in the entire column, i.e. along \columnwidth. As an alternative, you can make philex center the text within the philex text area itself. This puts the text slightly more to the right. To achieve this, give the command \narrowcenter before the relevant environment. To later revert to the default option, give the command \widecenter.

Alignment

Horizontal alignment within a philex environment can be easily achieved by means of Scott Pakin’s elegant eqparbox package. The code

\lb{alignthis}{\eqmakebox[hit][l]{\(2+3\)} quad = \quad \(5\)
\eqmakebox[hit][l]{\(2+3+4\)} quad = \quad \(9\)}

automatically generates alignment after a second round of typesetting, without the need of an equation align environment:

\begin{equation}
\begin{aligned}
2 + 3 &= 5 \\
2 + 3 + 4 &= 9
\end{aligned}
\end{equation}

Labelled brackets

linguex provides the \exi. command for generating labelled brackets that display syntactic structure. This command can be used in philex but can not be integrated into proper philex commands without loss of philex features, such as hyperlinks. However, it is easy
to add labels manually, especially by means of the \textsubscript command provided by the fixlt2e package, by Frank Mittelbach, David Carlisle, Chris Rowley, and Walter Schmidt. The result will be the same. The following example is from the linguex manual but implemented with \textsubscript:

\lb{braclabel}{[[\textsubscript{NP} Fritz ][ snores ]]% \textsubscript{S}}

(4) [[NP Fritz ][ snores ]]S

Glosses

linguex also provides the the \textsubscript{exg} command, and associated sub-sentence commands, for generating glosses. As above, these commands are available when philex is used, but cannot be integrated in philex environments without loss of features. However, linguex here builds on the cgloss4e package by Hans-Peter Kolb and Craig Thiersch. This package is called by philex and the commands of cgloss4e can be directly applied in philex environments. The \textsubscript{gll} command introduces the sentence-gloss pair, and the \textsubscript{glt} or \textsubscript{trans} command introduces free translation. The following example, here embedded in \lb, is taken from the manual for cgloss4e (which is also the manual for gb4e):

\lb{gloss}{\textsubscript{gll} Wenn jemand in die Wüste zieht ... \textsubscript{trans} 'if one retreats to the desert and ... '}

(5) Wenn jemand in die Wüste zieht ...
    'if one retreats to the desert and ... '

Grammaticality judgments

Grammaticality judgments, and other judgments of oddity, can now be inserted, with the symbols of your choice, by means of the \textsubscript{oddity} command.

\lb{gram}{
  \lba{grama}{\textsubscript{oddity}{*}He lives in N.Y since two years.}
  \lbb{gramb}{\textsubscript{oddity}{??}Whom does he live in N.Y. since two years?}}

delivers
(6) a. *He lives in N.Y since two years.
b. ??Whom does he live in N.Y. since two years?

Top-level brackets

By means of philex commands, the brackets can be removed or replaced for all the top-level environments except \lb itself. However, as of version 4.0 of linguex, the brackets for the \lb command can be customized.

\renewcommand{\ExLBr}{\}}
\renewcommand{\ExRBr}{[}
\renewcommand{\FnExLBr}{\{}
\renewcommand{\FnExRBr}{\}}

You can change the \lb brackets to square brackets in main text and to curly brackets in footnotes.

For the other cases, you might e.g. want an \lbp environment without name. To turn the brackets off, give the command \broff before the environment. To turn them back on later, give the command \bron.

The non-\lb top-level brackets can be reset by the two-argument command \philbrackets. For instance \philbrackets{[}{\}} will make the following environment names have square brackets.

Punctuation

There is sometimes a reason to embed the content of an environment in running text or in another environment, so that changes in the original environment are automatically reflected (see below on cross-references). You might then not want the original full stop of an environment sentence to come along. For this purpose, there is a punctuation command \philpunct at the end of top-level environments. By default, \philpunct is set to be empty. You can redefine it to provide a full stop by means of \philfullstop before the environment. The command \philcomma redefines it to produce a comma, \philexclaim produces an exclamation mark, \philquestion a question mark, and \philempty turns it back to empty. By means of \renewcommand{\philpunct}{\} you can set it as you like.

In earlier versions of philex the corresponding commands were \p, \s (for full stop), \km (for comma) and \q (empty). Mostly because I have learned that it is bad design to use one-letter commands, these are now replaced by the new commands. In case of typesetting a file with the old commands, choose the package option <oldpunct>.
linguex has sensitivity to footnote contexts, and provides a separate counter, \texttt{FnExNo}, that is reset in each footnote. This is inherited by philex.\footnote{\texttt{linguex} has sensitivity to footnote contexts, and provides a separate counter, \texttt{FnExNo}, that is reset in each footnote. This is inherited by philex.}

## 4 Sub-environments

\texttt{lba}, \texttt{lbb}, and \texttt{lbz} \par
\texttt{lba}, \texttt{lbb}, and \texttt{lbz} provide sub-environments (sublists).

To illustrate, the code

\begin{verbatim}
\lbp{clauses}{PP}{Some main words, followed by
\lba{first}{Time flies
\lba{firstnew}{Like an arrow
\lbz{lastnew}{And much too fast}}
\lbb{second}{But never stops}
\lbz{last}{Which is lucky}
and a concluding comment.}
\end{verbatim}

produces

\begin{verbatim}
(PP) Some main words, followed by
  a. Time flies
    (i) Like an arrow
    (ii) And much too fast
  b. But never stops
  c. Which is lucky
and a concluding comment.
\end{verbatim}

As shown in the source example, the sub-environments must be embedded in the last argument of the top-level command. A second-level sub-environment need not be embedded in the first-level sub-environment (and so on).

As exemplified, sub-environments can be nested up to three levels. The first two levels have their own counters, \texttt{SubExNo} and \texttt{SubSubExNo}, provided by \texttt{linguex}.\footnote{The result is illustrated by}

\begin{verbatim}
(i) A first footnote example:
(ii) A second footnote example.
\end{verbatim}

which shows that numbering is lowercase roman.
The \texttt{\textbackslash lba} command restarts the counter of the relevant level. The \texttt{\textbackslash lbb} command steps the counter, while the \texttt{\textbackslash lbz} command steps the counter and closes the level, returning to the next higher level. It is not necessary to use \texttt{\textbackslash lbz} for the last item at the first sub-level (i.e. at the end of a top-level environment). In case only a single sub-environment is used, the command \texttt{\textbackslash z} of \texttt{linguex} returns to the higher level.

The \texttt{name} of the first two sub-levels can be customized in two ways, with respect to numbering style and with respect to brackets. The command \texttt{\textbackslash subformat} takes three arguments.

The first argument determines the numbering style. The value ‘R’ displays the counter in upper-case roman numerals, ‘r’ in lower case roman, ‘1’ in arabic numerals, ‘A’ in upper-case alphabetic listing, and ‘a’ in lower-case alphabetic listing. Default for the first sub-level is lower-case alphabetic.

The second argument sets the prefix (opening bracket) and the third argument sets the suffix (closing bracket). Default for the first sub-level is no left-hand bracket and a period for the suffix, as shown in the example.

The corresponding command for the second sub-level is \texttt{\textbackslash subsubformat}, with the same arguments and values. Default for the second sub-level is lower-case roman numerals for displaying the counter, and ordinary left- and right-hand round brackets, as shown in the example (these defaults are set by \texttt{linguex}). To exemplify, the source code

\begin{verbatim}
\lbu{clausesup}{clauses}{$'$}{
\subformat{A}{}{)}
\subsubformat{1}{[}{]} Some introductory words, followed by
\lba{firstb}{Time flies}
\lba{firstnewb}{Like an arrow}
\lbz{lastnewb}{And much too fast}
\lbb{secondb}{But never stops}
\lbz{lastb}{Which is lucky} and a concluding comment.}
\end{verbatim}

produces

\begin{verbatim}(PP') Some introductory words, followed by
A) Time flies
\end{verbatim}
Like an arrow
And much too fast
B) But never stops
C) Which is lucky
and a concluding comment.

If the sub-formatting commands are given within the top-level environment, they control only the labels within that environment. If they are given outside, the commands are valid until superseded by later commands.

Names for sub-environments

The sub-level commands have an optional argument (inherited from linguex) that can be used for naming sub-environments. For instance, the source

\lb{crane}{The saying
\lba[S]{squeak}{The squeaky wheel gets the grease}
\z.

is often confirmed.}

produces

(7) The saying
(S) The squeaky wheel gets the grease
is often confirmed.

Round brackets are put in by default. This can be reset locally by means of the \philbrackets command. For instance:

\lbu{craneup}{crane}{\('\)}{\philbrackets{[]}{[]}}The saying
\lba[S]{craneupa}{The squeaky wheel gets the grease}
\z.

is often confirmed.}

(6') The saying
[S] The squeaky wheel gets the grease
is often confirmed.

Punctuation for sub-environments

Punctuation for sub-environments is controlled by means of the commands \philsubpunct, \philsubstop, \philsubcomma, \philsubexclaim,
\texttt{\textbackslash philsubquestion}, and \texttt{\textbackslash philsubempty}, analogous to the corresponding top-level commands. The corresponding older commands where \texttt{\textbackslash pt}, \texttt{\textbackslash stp} (full stop), \texttt{\textbackslash kmt} (comma), and \texttt{\textbackslash qt} (empty). As before, they are made active by means of the \texttt{\textless oldpunct\textgreater} package option.

5 Cross-references

The first (mandatory) argument of all environment commands are used by \texttt{philex} to give a \texttt{\textbackslash label} command. If the environment is numbered by a counter, the label makes the environment accessible for cross-reference by the standard cross-reference mechanism of \LaTeX{}. For instance, \texttt{\textbackslash ref\{nice\}} produces the cross-reference ‘1’ to the first \texttt{\textbackslash lb} environment on page 2.\footnote{Cross-reference on page ??}

However, \texttt{philex} provides its own cross-reference command, \texttt{\textbackslash rf}, which has a wider function. It provides ordinary cross-references in accordance with the numbering style of the antecedent, and keeps track of the embedding. It also provides cross-references to named environments, where there is no counter value to pick up. The difference between \texttt{\textbackslash rf} and \texttt{\textbackslash rn} is that the latter leaves out the brackets. So we will have the productions

\begin{tabular}{ll}
\texttt{\textbackslash rf\{nice\}} & \texttt{\rightarrow (1)} \quad \text{(target on page 2)} \\
\texttt{\textbackslash rf\{kne\}} & \texttt{\rightarrow ((T3))} \quad \text{(target on page 4)} \\
\texttt{\textbackslash rn\{kne\}} & \texttt{\rightarrow (T3)} \quad \text{(target on page 4)} \\
\texttt{\textbackslash rn\{clauseup\}} & \texttt{\rightarrow PP'} \quad \text{(target on page 8)} \\
\texttt{\textbackslash rf\{lastnew\}} & \texttt{\rightarrow (PPaii)} \quad \text{(target on page 7)} \\
\texttt{\textbackslash rf\{lastnewb\}} & \texttt{\rightarrow (PP'A2)} \quad \text{(target on page 8)} \\
\texttt{\textbackslash rn\{lastnewb\}} & \texttt{\rightarrow } \quad \text{(target on page ??)} \\
\texttt{\textbackslash rf\{lastnewb\}} & \texttt{\rightarrow ()} \quad \text{(target on page ??)}
\end{tabular}

Both \texttt{\textbackslash rf} and \texttt{\textbackslash rn} have an optional argument which produces an optional suffix. \texttt{\textbackslash rf\{\ast\}\{nice\}} produces (1*). The last example is not perfect, as the round brackets are produced instead of the square brackets used at the target. As things stand at the moment, this has to be corrected manually by using \texttt{\textbackslash rn} instead, or \texttt{\textbackslash rxn}.\footnote{Cross-reference on page ??}
When cross-references to subsentences are local, i.e. appear shortly after the top-level sentence, you might find it inelegant to include the main name or number. In such cases, context disambiguates an abbreviated cross-reference. The commands `\rfx` and `\rnx` allow you leave out its head. Compare with lines five and six of the list above:

\rfx{lastnew} \rightarrow (aii) \hspace{1cm} \text{(target on page 7)}
\rnx{lastnewb} \rightarrow A2 \hspace{1cm} \text{(target on page 8)}

Both `linguex` and `philex` provide customizing of inner delimiters in complex cross-references. `linguex` has changed in version 4.0 (see its manual). With `philex`, use the `\phildashes{}{}` command. This works together with the `\subformat` and the `\subsubformat` commands. The first argument sets the delimiter between the top-level reference and the reference to the first sub-environment, while the second argument sets the delimiter between references to the first and the second sub-level environments. To illustrate. We use the input

\lb{cool}{We would like a colon}
\phildashes{:}{}
\subformat{a}{i}{}
\lba{coola}{before reference to this.}
\lba{coolb}{but nothing more before reference to this}

\vspace{1cm}

(8) \text{We would like a colon}
\hspace{1cm}
\begin{itemize}
\item[a)] \text{before reference to this}
\item[i)] \text{but nothing more before reference to this.}
\end{itemize}

If the `hyperref` package is called (must be called last) in the preamble, all ordinary cross-references of the label+cross-reference mechanism of LaTeX are by default made into pdf hyperlinks. This includes all cross-references in `philex` that rely on counters. If the `<hyper>` package option is also called, cross-references to named environments are made into hyperlinks as well.

\footnote{The reason for the fuss about brackets is that in order to modify the material within the brackets, e.g. with \texttt{\lb{bu}}, the brackets cannot be hard-coded into the cross-reference command, but must be put in afterwords.}
This is useful for reading pdf:s on the screen, and also for beamer slide-show presentations: click on a cross-reference to a named principle, and you are taken to the page where that principle is stated. In beamer you are taken to the first slide of the relevant frame.

Sub-environments can be used in footnotes as well. But here, to get the cross-references right, the subformat command must be used. The example is produced by the input

\lb{ffnote}{A first footnote example:}  
\subformat{a}{()}  
\lba{fnota}{Be careful in footnotes.}  
\lb{sfnote}{A second footnote example. This cross-reference to \rf{fnota} is correct.}

philex also has cross-reference functions for repeating the contents of environment. By means of \rp{nicy} you repeat the content of (1'): But yesterday was even nicer. If you want to embed the content in another sentence, you may want the initial capital letter to be made lowercase. Then use the \ml command, as in

(9) John says that this is a nice day.  
    Not too hot.

produced by

\lb{jsays}{John says that \ml{nice}}

In case you want a full stop at the end of an earlier environment and want to embed the content without the stop, you can use the \philfullstop command to have it inserted for you:

\philfullstop  
\lb{stop}{It is very cold}  
\philexclaim  
\lb{ifstop}{If \ml{stop}, you should put on a cap}

\footnote{This is illustrated by}

(i) A first footnote example:  
    a) Be careful in footnotes.  
(ii) A second footnote example.  

This cross-reference to (ia) is correct.
produces

(10) It is very cold.
(11) If it is very cold, you should put on a cap!

One can repeat an entire environment with label by means of the \rff command. \rff{nice} produces

(1) This is a nice day.
   Not too hot.

\rffnot

   Occasionally, a named or numbered sentence carries a footnote reference, and when repeating the sentence later, you may not also want to repeat the footnote reference, nor the note itself. The alternative command \rffnot allows you to leave them out:

\lb{norepeat}{This ends with a footnote reference.\footnote{You don’t want to repeat this note.}}

(12) This ends with a footnote reference.\footnote{You don’t want to repeat this note.}

Now you can repeat the sentence without the note by \rffnot{norepeat}, which delivers:

(11) This ends with a footnote reference.

6 Lengths and spaces

\linguex provides a number of length and spacing commands that can be used to customize the appearance of \philex environments. Below are given the length name, brief explanation, and the default value set by \linguex. ‘hspace’ is short for ‘horizontal space’ and ‘vspace’ analogously:

\Exlabelsep (hspace after label) 1.3em
\Extopsep (extra vspace above and below) .66\baselineskip
\SubExleftmargin (hspace after sub-level name) 2em
\SubSubExleftmargin (hspace after sub-sub-level name) 2.4em
\Exindent (indent of the environment) 0pt

\footnote{You don’t want to repeat this note.}
Beside these, \texttt{philex} modifies \texttt{linguex} by letting two re-definable commands replace hard-coded lengths:

\begin{itemize}
  \item \texttt{pclabelwidth}{} (width of label) \hfill 3em
  \item \texttt{pclabeldefault} (returns to default value)
  \item \texttt{pclabelsep}{} (sets hspace after label to the argument, e.g. 1cm)
  \item \texttt{pclabelsepd} (sets default value of the label separation)
  \item \texttt{Exredux} (vspace reduction between environments) \hfill \texttt{-\baselineskip}
\end{itemize}

As of version 1.3, \texttt{philex} has built-in features for automatic handling of long names. By default, the \texttt{\labelsep} parameter adjusts so that the inner left margin moves to the right. This is illustrated by:

\begin{verbatim}
\broff
\lbp{compo}{Compositionality:}{The meaning of a complex expression is a function of the meanings of its parts and its mode of composition.}
\end{verbatim}

Compositionality: The meaning of a complex expression is a function of the meanings of its parts and its mode of composition.

The \texttt{\lbp} and \texttt{\lbu} have this feature. For two-column documents, the label separation should preferably be somewhat reduced. This has to be set manually by means of these commands, since there is no automatic adaptation to two-column options or environments.

You might have several consecutive environments with names of different lengths that you wish to have horizontally aligned. This is automatically achieved by the \texttt{\lbpsep} and \texttt{\lbusep} environments. To illustrate:

\begin{verbatim}
\lbp{shortname}{Name}{You might have several consecutive environments with \texttt{names} of different lengths that you wish to have horizontally aligned.}
\lbpsep{longname}{shortname}{Longername}{You might have several consecutive environments with \texttt{names} of different lengths that you wish to have horizontally aligned.}
\end{verbatim}

produces
You might have several consecutive environments with names of different lengths that you wish to have horizontally aligned.

The \bpsep takes four mandatory arguments, where the second is the label of the philex environment you want to align with. You can add as many \bpsep environments as you like; if they have the same target (second mandatory argument), they will align to each other, so that the longest name sets the inner left margin for all others. In case a later environment has a longer name than an earlier one, as in the example above, an extra round of typesetting is required to achieve alignment.

The \busep command takes five mandatory arguments. The first three are as in the \bu command, the fourth provides the target of alignment, i.e. the label of the environment you want to align with (this might and might not be the same as the environment whose label you provide in the second argument), and the fifth provides the content of the environment.

In addition, philex provides a non-printing command \sepset whose only purpose is to provide a name for later alignment:

\sepset{labelline}{nameline}

After such a command has been given, labelline can be used in the third and fourth arguments of \bpsep and \busep, respectively.

As of version 1.3 of philex, the \bpx environments also have automatic margin adjustment for long names. In these cases, the name is set as usual, with the \bpxformat command. If the second or third argument is long, the inner left margin will adjust, as in:

\bpxformat{1}{\bf Definition }{.}
\bpx{firstdef}{A proposition is known \emp{a priori} to be true iff the knowledge is not evidentially based on experience.}

yields

**Definition 1.** A proposition is known *a priori* to be true iff the knowledge is not evidentially based on experience.
There are also starred variants of \lbp, \lbu, and \lbpx. The starred commands produce an indented first line in response to a long name, but leaves the left margin in place for the following lines:

\lbp*{compo2}{Compositionality}{The meaning of a complex expression is a function of the meanings of its parts and its mode of composition.}

produces

(Compositionality)  The meaning of a complex expression is a function of the meanings of its parts and its mode of composition.

Vertical spacing

Vertical spacing is controlled with \Extopsep and \Exredux. The second controls the reduction of inter-environment vertical space and needs to be adapted to the general control of vertical space. To illustrate, the input

\setlength{\Extopsep}{2\baselineskip}
\setlength{\Exredux}{-3\baselineskip}

has the following effect on vertical spacing:

(13)  First vertical space example
(14)  Second vertical space example
(15)  Third vertical space example

Next line comes here.

7 Troubleshooting

Infelicitous labels  The most common problem you can run into with philex derives from an infelicitous choice of label (first argument to the commands). philex creates new command names (control sequences) from the labels, and especially with short labels it sometimes happens that a generated label coincides with a control sequence defined by TeX, or LaTeX, or some document class or package. Typesetting will halt
with an error message that does not give the appropriate information. The location of the error need not be right either.

To remedy, try replacing your latest label(s) by new ones that are unlikely to clash with other control sequences.

Another type or error can occur in case the \texttt{hyperref} package is used, and \texttt{hyperref} does not find the information it needs in the \texttt{.aux} file, perhaps because of an error in a previous round.

The remedy is to enter non-stop typesetting mode by pressing ‘r’ in response to the error message. Typesetting will be completed, with incorrect results, but the next round of typesetting will then work as it should. If there is still a problem, abort typesetting, trash the auxiliary file and start over.

A problem may arise because LaTeX style math environment delimiters (\texttt{(...)}) are used in the third argument (the update symbol) of the \texttt{\lbu} command. To avoid this problem, use TeX style delimiters ($..$) instead.

A further possible problem derives from a clash between \texttt{linguex} and fairly recent versions of \texttt{hyperref}. This (as has been explained to me by Heiko Oberdiek) depends on a conflict over the use of \texttt{\b}. The result will be that sub-environments will lack labels and be incorrectly formatted.

The remedy has been provided by Heiko Oberdiek, who has added a workaround in \texttt{hyperref}. To make use of this, make sure that your version of \texttt{hyperref} is 6.78o (from 2009.02.02) or later.

There is a conflict between the \texttt{setspace} package and \texttt{linguex}. The problem seems to be that \texttt{setspace} redefines \texttt{@footnotetext} in a way that interferes with the way \texttt{linguex} is meant to work in footnotes.

To avoid this problem, load \texttt{philex} (and \texttt{linguex} too, if called) \texttt{after setspace}. Also, don’t use the \texttt{\setstretch} command in the preamble for setting line spacing. Rather set a \texttt{spacing} environment in the main body, after \texttt{\begin{document}}. This environment may span the entire main document.

There is a clash between \texttt{linguex} and the \texttt{SIUnitx} package. Thanks to Tom Hodgson and Alan Munn for the following description and solution:

\texttt{\texttt{linguex} defines a command \texttt{\bg} to introduce glosses. It also creates aliases to this command \texttt{\cg}, \texttt{\dg}, \texttt{\eg}, and \texttt{\fg}. The \texttt{\fg} command conflicts with \texttt{SIUnitx}, which defines \texttt{\fg} as fem-}
togram. The simple solution is to load linguex and then redefine \fg. to nothing, and then load SIUnitx. Since the \fg. command isn’t necessary in linguex (and isn’t even really documented), it’s an easily worked around problem, and not really a bug, since both are end user commands.

\documentclass{article}
\usepackage{linguex}
\def\fg{}
\usepackage{siunitx}
\begin{document}
Foo
\end{document}''

Clash with tipa There is also a clash between linguex and the tipa font/symbol package, as well as between tipa and philex with the oldpunct package option. Call tipa before calling linguex or philex (thanks to J. ‘mach’ wust). This does not, however, completely remove the conflict with the oldpunct package option. To avoid this, use current philex commands, which makes the option dispensable.

Comments Finally, bug reports, comments or suggestions are welcome and should be directed to Peter Pagin at peter.pagin@philosophy.su.se.

8 Version history

Changes in version 1.2

– Grammaticality judgments added.
– Bugfix: Horizontal spacing with \rn. Fixed.\footnote{This bug was fixed in version 1.01, but stupidly reintroduced in version 1.1 because of bad file management.}

Changes in version 1.1

– Local cross-references added.
– Possibility of repeating a sentence without its associated footnote.
– \bpxformat now resets bpx counters.
– Extra counter bnp and associated commands.
– Bugfix: cross-references to sub-sentences did not work properly in connection with unusual combinations of using and not using optional arguments. Now fixed.

Changes in version 1.01
– Bugfix: eliminated unwanted space after \rn{} cross-references.

Changes in version in version 1.0
– Adapted the package to version 4.1 of linguex.
– Added two commands for label separation.

Changes in version 0.9 (internal version)
– Added support for hyperref for named environments.
– Added naming options for sub-environments (\lba, \lbb, \lbz).
– Added customizing of brackets and numbering-style for second-level sub-environments.
– Corrected errors in vertical spacing for \lbu and \rff.
– Corrected an error in the centering option, and added an alternative for centering.
– Replaced earlier (overly short) punctuation commands, and added a package option for using the old commands.
– Added customizing of vertical spacing and horizontal spacing as well as label length.
– Made the \lbpx environments labels more customizable.
– Rewrote and reorganized the manual.

Changes in version 0.6
Corrected vertical spacing between adjacent \lb and \lbp environments.

Changes in version 0.5
In version 0.3, forward cross-reference was effected by means of writing command definitions to a separate file, myfile-px.tex, which was
subsequently read-in at the beginning of the following run of typesetting. In version 0.5 the information is instead written to the .aux file and then read from the .aux file in the subsequent run. This method is more elegant and eliminates problems with the implementation of the old method. The improvement is due to Robin Fairbairns.