Preliminary documentation for gb4e.sty*and cgloss4e.sty

Hans-Peter Kolb
Craig Thiersch <c_thiersch@alum.mit.edu>

Contents

1 Linguistic examples in L_{\LaTeX}  \hspace{1cm}  2
   1.1 Examples with running numbers \hspace{1cm}  3
   1.2 Examples with other identifiers \hspace{1cm}  4

2 Glosses  \hspace{1cm}  6

3 Varia  \hspace{1cm}  8

Important note:

The file gb4e.sty\footnote{Parts of “Government-Binding style” gb4e.sty are loosely based on ideas and code by M. Covington, M. van der Goot, J. Frampton and L. Holt. In particular it automatically includes cgloss4e.sty, a (heavily adapted) version of M. Covington’s \LaTeX-adaptation of M. van der Goot’s plain\TeX Midnight gloss macros.} allows _ (subscript) and ^ (superscript) to be used in ordinary text, which is handy, but because it changes their definitions it must be loaded after any file that uses them in their \TeX meaning.

\footnote{This version of gbz.sty redesigns the entire series of \texttt{\_ex} commands. It also contains changes to the gloss-macros to make them work in footnotes etc. \textbf{NB: The export version does not provide a soft-landing system of command-compatibility with earlier versions.}}

NB: Make sure to get cgloss4e.sty together with gb4e.sty.

This version of gb4e includes some minimal bug-fixes and enhancements by Alexis Dimitriadis, with emphasis on backward compatibility.
1 Linguistic examples in \LaTeX

gb4e.sty is a style option to be used with \LaTeX’s major document styles. To make its commands available, add it to your document’s preamble like this:

(in \LaTeX\texttt{2e}:) \texttt{\usepackage{gb4e}}

(in \LaTeX\texttt{2.09}:) \texttt{\documentstyle[...gb4e]{...}}

Its core is an environment for the formatting of linguistic examples.

The format is simple:
(Batches of) examples are introduced by \texttt{\begin{exe}} which sets up the top level example environment. Obviously, this environment has to be closed after the example(s) by \texttt{\end{exe}}. You can create sub- and subsubexamples by embedding several \texttt{xlist}-environments (up to three levels deep).

Here is an elaborate example:

\begin{exe} % sets up the top-level example environment
\ex\label{here} Here is one. % example with running number
\ex[*]{Here another is.} % judged ex. with running number
\ex Here are some with judgements.
\begin{xlist} % first embedding (alphabetical numbering)
\ex[*]{A grammatical sentence am I.}
\ex[??]{An ungrammatical sentence is you.}
\ex[?]{A dubious sentence is she.}
\end{xlist} % end first embedding
\ex Dare to judge me!
\end{xlist} % end first embedding
\ex This concludes our demonstration.
\end{exe} % end example environment

This produces:

1 (1) Here is one.
(2) * Here another is.
(3) Here are some with judgements.
   a. A grammatical sentence am I.
   b. * An ungrammatical sentence is you.
c. A dubious sentence is she.

d. i. ** Need one a second embedding?
       ii. % sometime.

e. Dare to judge me!

(4) This concludes our demonstration.

All example-list commands have an optional argument which allows you to set the labelwidth to, e.g., the widest label to come (in the style of \begin{bibliography}). So, \begin{exe}[(234)] will set the labelwidth to the width of “(234)”—which is also the default. The other defaults are “m.” for xlist, and “iii.” for xlisti. The default labelwidth for the exe-environment can also be changed globally by issuing a \exewidth{<string>} command in the preamble of your document (i.e. between \documentclass and \begin{document}).

1.1 Examples with running numbers

As just shown, every example with a running number is introduced by \texttt{\textbackslash\texttt{ex}}. Obviously, \texttt{\textbackslash\texttt{ex}} is context-sensitive: at the top level the number produced is \texttt{arabic} and accumulative throughout the document; in embedded xlist-environments it is \texttt{alphabetic} on the first level, \texttt{roman} on the second, and \texttt{arabic} on the third, starting with a., i., and 1., respectively. For total control over the shape of your subexample-numbering there are some special forms listed below:\footnote{Note that for all \texttt{\begin{env} \ldots \end{env}} pairs the forms \texttt{\begin{env} \ldots \end{env}} are also permissible in \LaTeX. So if you like you can go on using \texttt{\lstlist} for \texttt{\begin{lstlist} \ldots \end{lstlist}} etc., but \LaTeX-syntaxcheckers and utilities like AUC-\TeX\ won’t recognize them as necessarily paired and won’t report unmatched expressions. (Have you ever tried \texttt{\textbackslash\texttt{Ctrl}+C \texttt{Ctrl}+C} (= \texttt{LaTeX-environment}) or \texttt{\textbackslash\texttt{Ctrl}+C \texttt{Ctrl}+F} (= \texttt{LaTeX-close-environment}) in EMACS’ AUC-\TeX\ mode? You should—it saves a lot of typing.)}

<table>
<thead>
<tr>
<th>Environment</th>
<th>Default numbering</th>
</tr>
</thead>
<tbody>
<tr>
<td>\begin{exe} \ldots \end{exe}</td>
<td>(1) arabic (Toplevel)</td>
</tr>
<tr>
<td>\begin{xlist} \ldots \end{xlist}</td>
<td>depends on level of embedding</td>
</tr>
<tr>
<td>\begin{xlista} \ldots \end{xlista}</td>
<td>a. alphabetical</td>
</tr>
<tr>
<td>\begin{xlisti} \ldots \end{xlisti}</td>
<td>ii. roman</td>
</tr>
<tr>
<td>\begin{xlistn} \ldots \end{xlistn}</td>
<td>3. arabic</td>
</tr>
<tr>
<td>\begin{xlistI} \ldots \end{xlistI}</td>
<td>IV. Roman</td>
</tr>
<tr>
<td>\begin{xlistA} \ldots \end{xlistA}</td>
<td>E. Alphabetical</td>
</tr>
</tbody>
</table>

In all cases, \texttt{\textbackslash\texttt{label}}/\texttt{\textbackslash\texttt{ref}} commands will do The Right Thing.

The syntax of \texttt{\textbackslash\texttt{ex}} is as follows:

\footnote{Note that for all \texttt{\begin{env} \ldots \end{env}} pairs the forms \texttt{\begin{env} \ldots \end{env}} are also permissible in \LaTeX. So if you like you can go on using \texttt{\lstlist} for \texttt{\begin{lstlist} \ldots \end{lstlist}} etc., but \LaTeX-syntaxcheckers and utilities like AUC-\TeX\ won’t recognize them as necessarily paired and won’t report unmatched expressions. (Have you ever tried \texttt{\textbackslash\texttt{Ctrl}+C \texttt{Ctrl}+C} (= \texttt{LaTeX-environment}) or \texttt{\textbackslash\texttt{Ctrl}+C \texttt{Ctrl}+F} (= \texttt{LaTeX-close-environment}) in EMACS’ AUC-\TeX\ mode? You should—it saves a lot of typing.)}
\ex \implies a label with running number (e.g., if a sublist of examples follows).
\ex <example> \implies a numbered example.
\ex[<judgement>]{<example>} \implies a numbered example with judgement.

The space reserved for judgements can be controlled by the command \judgewidth{<string>}, which sets it to the width of <string> (Default: "??"). It can be issued in the preamble to change the default globally, or within an exe-environment for a local change.

**In short:** Use \ex for all numbered examples; if they have no optional argument (judgement) they will be flush left; if they have [ ] 's then there will be a space reserved for judgements. (NB: In this case you have to indicate the scope of the example itself with {...}’s.)

### 1.2 Examples with other identifiers

Sometimes it is necessary not to use running numbers but to specify the identifier of an example manually as in:

\ldots let us now contemplate Chomsky (1981:264, ex.29f):

(29) \alpha \lb{INFL AGR}\lb{VP} V\ldots

(30) (i) \beta \lb{VP} V-AGR\ldots
(ii) \beta \lb{VP} V-AGR\ldots \lb{NP}

For cases like this there is the command \exi{<identifier>} which works exactly like \ex:

\exi{<identifier>} \implies the identifier as label
\exi{<identifier>} <example> \implies the example labelled with identifier
\exi{<identifier>}{<judgement>}{<example>} \implies the judged example, labelled with identifier.

So the example above is produced by:

\ldots let us now contemplate Chomsky (1981:264, ex.29f):
\begin{exe}
\exi{(29)} \al\lb{INFL} AGR\lb{VP} V \ldots
\exi{(30)}
\end{exe}
\begin{xlist}[(ii)]
\exi{(i)} \be\lb{VP} V-AGR\ldots
\exi{(ii)} \be\lb{VP}\lb{VP} V-AGR\ldots\] NP\]
\end{xlist}
\end{exe}
\exi{can be used to deal with many labelling problems. The three most commons ones are predefined:

- Crossreferences (\texttt{\textbackslash exr\{<label>\}}):
  
  ...a sentiment already expressed in (1) on page 2, repeated here:

  \begin{exe}
  \textbackslash ref{here} Here is one.
  \end{exe}

- Crossreferences with primes (\texttt{\textbackslash exp\{<label>\}}):
  
  ...which contrasts sharply with

  \begin{exe}
  \textbackslash exp{here}[??]{Here is two.}
  \end{exe}

- Unnumbered examples (\texttt{\textbackslash sn}): 
  
  ...here we have an example

  \\% Colourless green ideas sleep furiously.

  of a sentence too famous to be numbered...

  \begin{exe}
  \sn[\%]{Colourless green ideas sleep furiously.}
  \end{exe}

  of a sentence too famous to be numbered\ldots

Again, these commands work exactly as \texttt{\textbackslash ex} or \texttt{\exi{<identifier>}}.
2 Glosses

gb4e.sty offers a set of commands for typesetting glosses based on M. Covington’s LATEX-adaptation of M. van der Goot’s Midnight gloss TEX-macros.\(^3\)

A gloss basically contains two elements:

\[ \text{\textbackslash gll (obligatory): introduces the actual sentence/gloss pair;} \]
\[ \text{\textbackslash glt or \textbackslash trans (optional): introduces a free translation.} \]

So the following

\begin{exe}
\ex
\gll Wenn jemand in die Wüst zieht ... \\trans 'if one retreats to the desert and ... '
\end{exe}

produces:

(5) Wenn jemand in die Wüste zieht und lebt dort von
    If someone in the desert draws and lives there from
    Heuschrecken oder sich im Wald verirrt hat und nährt
    grasshoppers or self in the woods errored has and nourishes
    sich von Wurzeln und Beeren ...
    self from roots and berries ...
    'if one retreats to the desert and lives there from grasshoppers or
    gets lost in the woods and lives off of roots and berries'

Since the macro counts words, to match one word with a multi-word gloss or vice versa, you either have to put the words in braces or use hyphens. For example, German zum could be glossed as either \{to the\} or to-the. And of course, if you have a word that is not glossed, like WH-trace \( e_i \) or a proper name, you have to tell the gloss line that there’s something there, e.g. with \{ \}.

(6) Den Fritz\(_1\) habe ich _-1 zum Essen eingeladen.
    the fred have I to the eating invited.
    I invited Fred for dinner.

\(^3\)They are defined in the file cgloss4e.sty which is automatically read by gb4e.sty.
\(^4\)There is also a three-line version of \textbackslash gll, \textbackslash glll for special morphological/phonological information (not for the free translation!).
is produced by

\begin{exe}
\ex
\gll Den Fritz_1 habe ich \_\_\_{}\_ zur Essen eingeladen.\\
\text{the fred have I {} {} to the} eating invited.\\\n\glt I invited Fred for dinner.
\end{exe}

Although it is possible to treat an optional judgement just as the first word of the sentence with a null-translation as in (7), the standard way of using an optional argument with \ex as in (8) is simpler (and conceptually more consistent).

(7) * Dit is een voorboeldje.  
   This is an example.  
   ‘Here’s where the translation would be’

(8) * Dit is een voorboeld.  
    This is an example.  
    ‘and this would be the translation line’

This output is produced by

\begin{exe}
\ex
\gll * Dit is een voorboeldje.\\
\{} This is an example.\\
\trans ‘Here’s where the translation would be’
\end{exe}

and

\begin{exe}[*]\{ 
\gll Dit is een voorboeld.\\
This is an example.\\
\trans ‘and this would be the translation line’
\end{exe}

respectively.

By default the glosses appear in “Linguistic Inquiry style” with all lines in \rm. To change a line (e.g. to italics if required for a particular journal), just put the appropriate lines in the preamble of your document, e.g., \let\eachwordone=\it.\footnote{Font commands that take an argument, such as \textit, can only be used with versions of gb4e after 2009/12/28. Upgrade to the latest version or use argument-less font commands, such as \it.} The same for \eachwordtwo and \eachwordthree
(used by the three line glosses introduced by \glll). Note that it will not work to put \it before the line in your text, as the words are parsed separately. For example, \let\eachwordone=\it yields

\begin{exe}
\begin{par} 
9) Und hier können wir sehen was für Unfug wird gemacht
and hier can we see what for nonsense will-be-made
wenn er einen ganz langen Satz binnen kriegt
if he a very long sentence inside gets.
‘and here’s what happens to a loooooong sentence’
\end{par}
\end{exe}

and \let\eachwordtwo=\sf gives

\begin{exe}
\begin{par} 
10) * Cette phrase n’a pas une traduction.
This sentence not has nought a translation.
\end{par}
\end{exe}

As you can see by these examples, you can also re-define the eachword variables in your file for individual examples by using the \let command within the \begin{exe} environment, and then the style change is local.

By default, the glosses will be single-spaced even in double-spaced environments. If you—or rather the journal you’re submitting to—don’t like that, just give the command \nosinglegloss in the preamble of your document.

3 Varia

\texttt{gb4e.sty} contains a few additional goodies (mostly abbreviations for frequently used constructions) which will be properly documented in a subsequent life. For now just some sample of uses:

- X-bar styles: Per default \obar{X} \ibar{X} \iibar{X} \mbar{X} yields $X^0 X \overline{X} X^{\text{max}}$. With primebars in the preamble, the outcome is $X^0 X' XP X^{\text{max}}$.

- Labelled bracketings: \lb{NP}[1]\lb{D}[1] the\lb{\iibar{N}} example yields $[\text{NP}_1 \text{D}_1 \text{the}\iibar{\text{N}} \text{example}]$. An analogous right-bracket (\rb) is also defined.

- Greek letters: the frequently used letters $\alpha\beta\gamma\delta$ and $\theta$ have abbreviations \al \be \ga \de \th which also work outside mathmode.

- Sub- & superscripts (\_ and ^) work outside mathmode, too.

\textbf{Note:} This feature is known to cause problems for a number of other packages. It is retained for backward compatibility. In case of problems, you can disable it by adding the command \noautomath in your
preamble immediately after loading the gb4e package. You can also re-enable it later with the command \automath. [Added 2009/12/28]

At the very end of the style-file a couple of special accents for “exotic” languages are defined. Also, check out the \attop-, \atcenter-macros (by Lex Holt) for aligning naughty (picture-)examples to their numbers, as well as J. Frampton’s movement-arrows:

\begin{exe}
\ex\atcenter{
\arrowalign{\fillright{\text{Did}}&\text{John } t_{\text{Agr}}&\text{ appear to be likely to win.}}
\atcenter{\fillright{\downarrow}}
\atcenter{\fillleft{\uparrow}}
\end{exe}