

paralist*

Some New List Environments

Bernd Schandl
schandl@gmx.net

Printed on August 16, 2000

Abstract

This package provides some new list environments. Itemized and enumerated lists can be typeset within paragraphs, as paragraphs and in a compact version. Most environments have optional arguments to format the labels. Additionally, the L^AT_EX environments `itemize` and `enumerate` can be extended to use a similar optional argument.

1 Introduction

In a posting to `comp.text.tex` in May 1998 someone asked about the possibility of an enumerated environment that (a) can be used within paragraphs, (b) takes care of enumeration and (c) has items that can be referenced. Another posting mentioned the package `theapa` as a possible solution. Now that I was looking for that kind of environment and found those old postings I had a look at `theapa` and decided to take out the part about list environments and rewrite it a little bit.

Over time, compact versions of `enumerate`, `itemize` and `description` have been added and optional arguments for most environments make it possible to define a special way of formatting the labels.

2 Package Options

Certain parts of the package are only executed if appropriate options are specified.

`newitem` The L^AT_EX environment `itemize` will be redefined to have an optional argument to specify the format of the label. See Section 3.

`newenum` The L^AT_EX environment `enumerate` will be redefined to have an optional argument to specify the format of the label. See Section 3.

*Package version v1.9 of 2000/03/05.

`increaseonly` If an optional argument for one of the enumerated lists is used, `\leftmargin n`, i. e. the indentation of the `\item` text, is made just wide enough for the label. Therefore `\begin{enumerate}` and `\begin{enumerate}[1.]` have different indentation. If this option is specified though, `\leftmargin n` is never decreased.

`defblank` The two environments `inparablank` and `asparablank` will be defined. See Section 5.4.

`cfg` The configuration file `paralist.cfg` is loaded if it exists. (default)

`nocfg` The configuration file is not loaded.

3 Formatting the Labels

All the itemized and enumerated environments have optional arguments to specify the format of the labels. The following examples will only work if you loaded `paralist` with the options `newitem` and `newenum`.

Using the L^AT_EX standard classes, `itemize` uses the following symbols for the labels of the four list levels: `• – * ·`. If you want to change this for a particular environment just say something like

```
\begin{itemize}[$\star$]
```

and `★` will be used instead of the default symbol.

The optional argument of the enumerated environment is a little more complicated, but whoever has worked with David Carlisle's `enumerate` package can skip the rest of the section since exactly the same mechanism (and almost the same code) is used.

The tokens `A`, `a`, `I`, `i`, and `1` can be used in the optional argument to produce the counter with one of the styles `\Alph`, `\alph`, `\Roman`, `\roman` and `\arabic`.¹ These letters may be surrounded by any other string involving any other T_EX expression. However, the tokens `A a I i 1` must be inside a `{ }` group if they should not be taken as special. A few examples follow.

```
\begin{enumerate}[(i)]
```

produces the labels (i), (ii), (iii) ...

```
\begin{enumerate}[\example] a)
```

produces example a), example b), example c) ...

```
\begin{enumerate}[{A}-1]
```

produces A-1, A-2, A-3 ...

```
\begin{enumerate}[\bfseries {Item} I]
```

produces **Item I**, **Item II**, **Item III** ...

Note that in the last example `[\textbf{Item I}]` does not work because the special token `I` is inside a group.

The `\ref` command produces only the counter without the surrounding text, so in the examples above you would get `i`, `a`, `1` and `I` respectively if

¹The set of tokens can be extended. Look for `\pl@hook` in the code section.

you referenced the first item.

4 Defaults for Labels and Margins

If you want your lists labeled differently from the L^AT_EX default throughout your document it is a bit awkward to use the optional argument of the environments all the time. Therefore three macros are provided to define the labels and the left margins of the list environments.

Note that the macros defining the labels do *not* adapt the left margins if a wide label is specified because this may have side effects on other list environments. If you want that change you have to explicitly use `\defaultleftmargin`.

If in any of the following three macros an argument is empty then the according label or margin is left unchanged.

`\defaultitem` The default labels for itemized environments can be set by using the macro `\defaultitem` which gets four arguments. To get the L^AT_EX default labels say

```
\defaultitem{\textbullet}%
{\normalfont\bfseries \textendash}%
{\textasteriskcentered}{\textperiodcentered}
```

(which is of course silly because you don't need to do anything if you want to stick with the default labels). If you want a triangle (\triangleright) instead of the endash for level two just say

```
\defaultitem{}{\$ \triangleright $}{}{}
```

`\defaultenum` The labels for the enumerated lists are formatted with `\defaultenum` using the mechanism described in Section 3. The L^AT_EX default labels could be defined by

```
\defaultenum{1.}{(a)}{i.}{A.}
```

If you want capital Roman letters for level three just say

```
\defaultenum{}{}{I.}{}.
```

`\defaultleftmargin` To change the left margin of the lists use `\defaultleftmargin`. The length `\leftmargin n` specifies the indentation of a list of level n with respect to the list of level $n - 1$ or the surrounding text (if $n = 1$). The environments that use `\leftmargin n` are (at least) `enumerate`, `compactenum`, `itemize` and `compactitem` (maybe a few more that I am not aware of). The L^AT_EX settings could be defined by

```
\defaultleftmargin{2.5em}{2.2em}{1.87em}{1.7em}.
```

In `twocolumn` mode L^AT_EX uses a smaller margin for the first level which could be defined by

```
\defaultleftmargin{2em}{}{}{}
```

If some of your changes should appear in *every* document that uses `paralist`, put them in a file `paralist.cfg` which is read at the end of the package in case it exists (unless you specified the option `nocfg`).

5 New Environments

5.1 Enumerated Environments

asparaenum The environment `asparaenum` is an enumerated environment in which the items are formatted as separate paragraphs.

As an example, we use `asparaenum` within this paragraph.

1. Every `\item` is basically set as a separate paragraph. The second line is *not* indented (this is a feature, not a bug).

2. The next `\item` looks like this and is labeled.

The example was produced by the following piece of code:

```
\begin{asparaenum}
  \item Every ...
  \item The next ... \label{p11}
\end{asparaenum}
```

By saying `\ref{p11}` we get 2.

inparaenum The `inparaenum` environment formats an enumerated list within a paragraph, just like the one in the introduction.

The example in the introduction was set by the following commands:

```
... of an enumerated environment that
\begin{inparaenum}[(a)]
  \item can be used within paragraphs,
  \item takes care of enumeration and
  \item has items that can be referenced. \label{p12}
\end{inparaenum}
Another posting mentioned ...
```

By saying `\ref{p12}` we get c.

compactenum The `compactenum` environment is just a compact version of the standard `enumerate` environment. All the vertical skips are set to zero (actually they are adjustable, see Section 7).

5.2 Itemized Environments

asparaitem The `asparaitem` environment is very similar to `asparaenum`. It just uses symbols instead of enumerating the items. The environment has one optional argument which specifies the symbol. For an example see Section 6.

inparaitem Similar to `inparaenum` I added an environment `inparaitem` which also has an optional argument. I don't really know why anybody would use it, but I added it because of symmetry.

compactitem The `compactitem` environment is again just a compact version of the standard `itemize` environment with all the vertical skips set to zero. So by using this environment

- you can save some space and
- specify the symbol.

- Let me add a longer item so that you can see that we have a different indentation than in the `asparaitem` environment.

The code of the example above is

```
\begin{compactitem}[$\circ$]
  \item you can save some space and
  \item specify the symbol.
  \item Let me add ...
\end{compactitem}
```

5.3 Descriptive Environments

`compactdesc` The `compactdesc` environment is copied from the L^AT_EX standard classes with all the vertical skips set to zero. By the way, does anybody know why `description` has to be defined by the document class and is not defined in `ltlists.dtx`?

5.4 Blank Environments

Someone requested list environments that print their items as if there was no list. It seems that this makes entering structured data a little easier in LyX. Since not everybody needs these (odd) environments they are only defined if the package is loaded with the option `defblank`. The following two environments do not have optional arguments because there is no label to format.

`asparablank` Every item is formatted just as if it was a regular paragraph. If you want to use the optional argument of `\item` you have to add some white space at the end because `\labelsep` is set to zero. Use something like

```
\item[\textbullet\hspace{.5em}]
```

`inparablank` The items are set just as regular text. The “white space problem” mentioned in the last paragraph is handled automatically. If I didn’t tell you, you wouldn’t know that this paragraph is set using the following construction:

```
... are set
\begin{inparablank}
  \item just as ...
  \item The ...
  \item is handled ...
\end{inparablank}
If I didn't ...
```

6 Nesting Environments

All the environments can be nested just as the standard list environments. It’s probably not a good idea to call another list environment within a

`inpara...` environment, but why should anyone want to do this? The maximal nesting level is four, just as for the \LaTeX environments.

This paragraph is

- ★ an example for the usage of `asparaitem` and its optional argument,
- ★ and a demonstration that (i) you can use `inparaenum` within `asparaitem` and (ii) can still reference it.

The reference was in subitem (ii). The code of the last example is

```
\begin{asparaitem}[$\star$]
  \item an example ...
  \item and a demonstration that
    \begin{inparaenum}[(i)]
      \item you can use ...
      \item can still ... \label{p13}
    \end{inparaenum}
\end{asparaitem}
The reference was in subitem (\ref{p13}).
```

7 Fine-tuning

Ok, I already hear someone saying “Your compact lists are a nice idea, but I’d like to have it a little less compact.” Here is a solution. The following skips can be adjusted using `\setlength` and affect the spacing of the `compact...` environments. The names are chosen similar to the \LaTeX names, so I just copy the explanation from `ltlists.dtx`.

`\pltopsep`: Space between first item and preceding paragraph.

`\plpartopsep`: Extra space added to `\topsep` when environment starts a new paragraph (is called in `vmode`).

`\plitemsep`: Space between successive items.

`\plparsep`: Space between paragraphs within an item – the `\parskip` for this environment.

Actually, the two `...topsep` skips are added before *and after* the list.

The default value for all of them is 0pt. It is probably a good idea to define them depending on the font size if they are non-zero, i. e. using units such as `ex` or `em`.

8 Bugs and Wishes

No bugs ... that I know of.

Feel free to let me know about any problems, suggestions and wishes you have concerning this package and its documentation. Praise is welcome, too ;-)

The newest version of this package can always be found on CTAN or at <http://members.xoom.com/schandl/paralist/>.

9 Acknowledgments

I want to thank all the users who helped me with their comments finding bugs and extending the package. A big “Thank you” goes to David Carlisle, because there wouldn’t be any optional arguments for the enumerated environments without the code from his `enumerate` package. Some pieces of code of the `inpara...` environments are inspired by Mogens Lemvig Hansen’s `shortlst` package.