Paradigms: Headache?

Kees van der Laan

Hunzeweg 57, 9893 PB Garnwerd, The Netherlands
cgl@rc.service.rug.nl

1 BLUe’s Design II

Hi folks. When using \LaTeX I was plagued in the past by the little glob of white space on the first line after a head. It made me clear that ‘spaces are harmful,’ to paraphrase Dijkstra. When designing blue.fmt I had to look at the problem again among others, and to
- provide a simple suite, with hardly no knowledge of the user assumed about parameter separators
- gobble redundant spaces as much as possible
- allow also processing on the fly of the title
- choose a coding style, and
- choose a ‘markup language.’

My starting point was \beginsection The \TeXbook, p355. I chose head as root name conform to TUGboat.

2 Headers

I will design \beginhead and \endhead, next to the minimal markup variant \head{...}.

2.1 Design goals

From a typography point of view I would like to
- discourage to set the title alone at the end of a page
- typeset (flexible) vertical space (big) before and after (med) the title
- gobble spaces at the beginning of the title
- set the title in bold face (and the current size) undented
- don’t indent the first line of the text after.

2.2 What are the problems?

The markup language constructs have to be chosen. In macro expansion the two-part macro \TeXnique should be the basis. In blue’s format I adopted the pairs \begin<tag> and \end<tag>.

In order to set the title loose from the context we have to determine values for the glue, and be sure that it disappears at the top of a page. The coding solution below has different values for the parameters as supplied in \beginsection. From experiments it turned out that the values supplied by Knuth don’t give nice results with ‘BLUe’s Format.’ Too much glue was inserted for my taste.

Another problem is to prevent the headtitle to be printed at the bottom of a page on its own. The latter is related to the ‘widow-orphan’ phenomenon.

The coding of the minimal markup variant without parsing but with processing on the fly of ‘the argument’ has to be resolved in some elegant way.

The idea of a \pre<tag> and \post<tag> is general. It parameterizes the placement of a document element within context. Important! Whether you talk about titles, displays, tables, graphs, or you name it. Perhaps this should be generalized in something like \pasteup<tag>.

2.3 Take a deep breath

The minimal markup form, with the same functionality, reads

\beginhead# \begin{tag} \bgroup \aftergroup \endhead \afterassignment \ignorespaces \let\dummy=\ignorespaces

Explanation. The # as last character of the parameter text makes that the opening brace and what follows is placed after the replacement text of \head, The \TeXbook, p204. The replacement text starts a group via \bgroup and reads away the opening brace. The latter \TeXnique has been borrowed from plain’s \footnote, The \TeXbook, p.363, as used at the end of the replacement text of \fot. Spaces which precede the title are ignored via the invoke of \ignorespaces at the right time. Quite something isn’t it?

2.4 Paradigm

We should build upon two-part macros. In order to add the short variant to similar two-part macros the following template can be used.

\begin{tag}# \begin{tag} \bgroup \aftergroup \endtag \afterassignment \ignorespaces \let\dummy=\ignorespaces

\footnote{The subhead variants are similar but different in detail. Especially in undoing glue when heads follow-up each other without intervening text. Beware!}
2.5 Take a deeper breath

When using the above template in practice it occurred to me that \TeX could handle that too. For \head the desired functionality is that

\begin{verbatim}
\onepartmacro{head}
\end{verbatim}

will build the macro \head based upon the two-part macros \beginhead and \endhead.

The naive coding—which does not work, but makes the purpose quite clear—reads

\begin{verbatim}
def\onepartmacro#1{\def\#1##{\begin#1
   \bgroup\aftergroup\end#1
   \afterassignment\ignorespaces
   \let\dummy=}}
\end{verbatim}

Several desired functionalities have to be coded in \TeX's peculiar way, with as result

\begin{verbatim}
def\onepartmacro#1{\ea\let\csname begin#1\endcsname=0
   \ea\let\csname end#1\endcsname=0
   \toks0={\bgroup\afterassignment\ignorespaces
   \let\dummy=}
   \ea\xdef\csname#1\endcsname##{\csname begin#1\endcsname
   \the\toks0 \csname end#1\endcsname
   \the\toks1 }}
\end{verbatim}

Explanation. A lot of the trickery about partial expansion, The \TeXbook, p.216, has been applied. First a \csname is used to compose the names from the root name. Second the one-part macro is defined via an \xdef with suppression of the expansion of the \begin<name>, \end<name>, and the other tags. It is an \xdef to keep the auxiliaries used local. The first is handled via the property that a control sequence equal to a character (in general an unexpandable control sequence) is not expanded. The second is handled by the property that a \the on a token list is a one-step expansion.

2.6 Some remarks

Note that I did not gobble spaces at the end of the title argument. In the headtitle they won’t harm because the title is set on a line of its own.\footnote{In the other heads the space at the end of the argument can be there on purpose. Watch out!}

Processing a title on the fly is needed for verbatims and in general when we don’t like the ‘eyes’ of \TeX to set catcodes differently from those the script expects when digested in the ‘stomach.’\footnote{blue.fmt is available form the CTAN, as is the accompanying article BLUe’s Format.}

The good news

is that via this mechanism we can abstract at the user level from token variables and definitions. The user can supply the information as if it was a token variable. The using of the long form or the minimal form has been reduced to a matter of taste. And that is a good thing.

3 Tests

This note itself contains examples of use in its original form.

\begin{verbatim}
\beginhead|Head text in verbatim|\endhead
Text after next line
\end{verbatim}

with results

Head text in verbatim
Text after next line

Head text, with \texttt{this} in verbatim
Text after next line

Run it,\footnote{Run it, and have fun. All the best.} and have fun. All the best.

4 Note added in proof

In order to allow for a blank line after the heading it is necessary to replace \ignorespaces by a more general command like \ignorewhitespace due to Phil Taylor, or TUG's \ignorespacesandpars. A blank line after a heading is natural for a text without explicit markup. Knuth started from natural markup and stayed close to that, in MHO. His markup tags to format the chapter (titles) in the \TeXbook file are

\begin{verbatim}
\beginchapter 
\endchapter
\end{verbatim}

that is, imagine ASCII text with visual layout and in order to mark up the title and text insert the above commands, with the natural blank line already there to end the arguments of \beginchapter. Knuth's \beginchapter not only takes care of the title but also handles the chapter contents on the fly until \endchapter is endcountered. Really much different from Ka\TeX's approach to name but one general set of markup tags.

Knuth and natural markup.

Imagine that the ASCII text of a section reads

Section 1. Title

Now starts the section text proper....

And this is the last line of it.
What is the simplest and most natural markup for this?
First the requirements
- Section should be flexible towards language variation
- Title should be processed on the fly

A solution is the following extension heavily inspired by \beginchapter of manmac
\def\beginsection#1 #2#3. {\bgroup
  \def\par{\egroup\the\posthead}\
  \beginhead#1 #2#3. }
\posthead{\nobreak\medskip
  \noindent\ignorewhitespace}
\let\endsection\relax% or forget about it
\def\ignorewhitespace{\begingroup
  %Due to Phil Taylor EuroTeX 94
  \catcode\ =9
  \catcode\ÃÄ=9
  \catcode\ÃÈ=9
  \catcode\ÃÍ=9
  \futurelet\dummy\endgroup}
The marked up script is obtained by only inserting
\beginsection!!!
\beginsection Section 1. Title

Now starts the section text proper....
And this is the last line of it.
\endsection%for the purists

The beauty of the above approach, Knuth’s IMHO, is that the total section is handled, the title as well as the contents part, with the title proper processed on the fly, and that it remains as close to natural markup as I can think of. Language problems are absent. One can abstract from ‘Section’ of course via a toks variable \sectionname for example. Automatic numbering can be inserted too, but the question is whether it is really handy, especially in view of subsections and the like.

Acknowledgements.
Phil Taylor is kindly acknowledged for his \ignorewhitespace as generalization of \ignorespaces, and his suggestion to assign \relax to tokens instead of the number 0 to inhibit expansion.