The T\TeX\ Hierarchy

Donald Arseneau\textsuperscript{*}, Raymond Chen\textsuperscript{†}, Victor Eijkhout\textsuperscript{‡}

Introduction

For the UNIX operating system, a list of characterizations exists describing what constitutes a novice, a user, a guru, ... Here we give a similar list for users of \TeX. The reader is kindly asked to take this purely in a humorous vein.

The name

Novice says ‘tecks’.
User says ‘tecchhh’ but still moistens the screen doing it.
Programmer correctly pronounces ‘\TeX’.
Wizard has made at least one bad pun on the name \TeX.
Guru knows that even Knuth says ‘tek’.

The manual

Novice owns the TeXbook.
Programmer has just made a first correction to the text.
Hacker has formatted \texttt{texbook.tex} and knows about Knuth’s ‘little joke’.
Wizard is thinking of ways to supply the missing ‘tactile and olfactory sensations’ of \TeX.
Guru thinks ‘\TeX: the program’ is more useful.

The index of the \TeX\ book

Novice is confused by the number of references for each entry, has laughed at ‘\TeX: bad puns on the name’, and has counted the number of middle names of Barbara Beeton.
User knows about Bo Derek (in the \texttt{texbook} book), Jill Knuth (in the Metafont book), and Ellen Gilkerson (in the \LaTeX\ book).
Wizard knows why some entries are italicized or underlined.
Guru knows to look up Bourbaki for smart line breaks in paragraphs.

The system

Novice has found many bugs in \TeX.
User has learned that there are no bugs in \TeX, but doesn’t understand why ‘it doesn’t work!’.
Guru has actually found bugs in \TeX; frames the check from DEK.

Guru extraordinaire cashes checks from DEK.

Famous people

Novice is not sure whether Leslie Lamport is a man or a woman.
User knows not to capitalize ‘barbara beeton’.
Wizard knows how to pronounce ‘Knuth’ and ‘Eijkhout’.
Guru Knuth has asked about their middle name(s).

Programming style

Novice uses grouping without knowing why.
User writes \texttt{\textbf{... }} and doesn’t understand what went wrong.
Programmer writes \texttt{\def\bold#1{\textbf{#1}}).
Programmer first class writes \texttt{\long\def\bold#1{\textbf{#1}}).
Hacker writes \texttt{\def\beginbold{\bgroup\textbf} \def\endbold{\egroup}.
Wizard writes \texttt{\def\bold{\bgroup\textbf\let
ext=}).
Guru writes \texttt{\def\bold#1{\bgroup\textbf\let\next= }.

Style (cont’d)

Novice has heard of ties.
User inserts ties and writes ‘dr.\ ‘.
Hacker writes ‘dr.\ ‘, except in bibliographies where frenchspacing is in effect.
Guru Makes ‘.’ an active character in bibliographies so that ‘D.E. Knuth’ means ‘D.,E.\penalty300 Knuth’.

Errors

Pre-novice wonders why ‘Q’ takes so long to quit.
Novice will exit on the first ‘error’, even if the message starts with ‘OK’.
User keeps pressing return to scroll past errors, until that gets into an infinite loop.
Guru having written the input file with ‘\texttt{cat >} in the first place, the guru will type ‘i’ at an error, correcting all typos and supplying all missing mac-
Capacity Exceeded . . .

Novice constantly runs into the ‘TeX capacity exceeded’ error and asks the admin to build a larger version.

User knows how to find unbalanced curly braces.

Hacker occasionally runs into the ‘TeX capacity exceeded’ error and usually finds a way around them.

Wizard knows how to increase \TeX’s capacity, taking care to read DEK’s warnings about setting the values too high.

Guru ignores DEK’s warnings.

Printing and previewing

Novice prints the whole document after each run of \TeX.

User knows of previewers.

Programmer knows at least two previewers and vigorously argues why one is utter garbage.

Wizard thinks that
\begin{verbatim}
\tracingoutput=1
\showboxdepth\maxdimen
\showboxbreadth\maxdimen
\end{verbatim}
is the best previewer.

Macros

Novice has heard of macros, but has never seen one.

User writes macros that are used once, and that are longer than the code they replace.

Programmer having been bitten by unwanted spaces, writes macros that don’t contain spaces, and every line ends with a ‘%’.

Hacker has written self-modifying macros, writes \begin{verbatim}
\endlinechar=-1 \catcode`\M=9
\end{verbatim}
to prevent having to put ‘%’s at the end of lines in macros.

Guru has written macros containing #### more than 3 \begin{verbatim}
\expandafter\endcsname.item[Fossil]
\end{verbatim} in a row, and the sequence
\begin{verbatim}
\expandafter\endcsname. item[\text{Fossil}]
\end{verbatim} still has macros written in \TeX78.

Macros (cont’d)

Novice has written a macro \begin{verbatim}
\box\end{verbatim} to draw a box.

User has renamed it to \begin{verbatim}
\boxit\end{verbatim}.

Wizard has redefined \begin{verbatim}
\box\end{verbatim} so that it can have \begin{verbatim}
\verb\end{verbatim} in its arguments.

\LaTeX

Novice uses \LaTeX because colleagues and friends do.

User uses \LaTeX, even though colleagues and friends use Microsoft Word, or Word Perfect.

Wizard uses \LaTeX for journal and conference submissions, but homegrown macros when working alone.

\LaTeX errors

Novice actually takes the manual when it says ‘LaTeX error. See LaTeX manual for explanation.’

User knows what the relevant bits of \LaTeX error messages are.

Programmer knows what to type at the question mark when \LaTeX reports ‘\begin{verbatim}
\begin{document}
\end{itemize}\end{verbatim}’.

Wizard doesn’t make errors in \LaTeX, and answers questions about \LaTeX by editing latex.tex, lplain.tex, article.sty, or art10.sty.

\LaTeX style

Novice types a$_1$ because the error in a$_1$ occurred on the ‘_’.

User types \begin{verbatim}
a$_1$\end{verbatim} because Leslie Lamport says so.

Other packages

Novice could do more in Pagemaker.

User doesn’t see the difference between \TeX macros and Word Perfect macros.

Hacker writes macros to make \TeX look more like troff.

Wizard types \begin{verbatim}
\input troff\end{verbatim} to process old troff files.

Guru types
\begin{verbatim}
\input txtmacros
\input text.txt
\end{verbatim} to format plain text.

Life, everything

Novice thinks that learning \TeX will take a long time.

User realises that it wasn’t so bad after all.

Programmer tries to convince himself that the next macro is really going to save time in the future.

Wizard daydreams idly about how much he could have done with his life if he had never heard of \TeX.

Guru realises that a life without \TeX is not worth living.

(Also thanks to Barbara Beeton, Tim Chow, Denys Duchier, Dan Ellard, Michael Sofka.)