Which Success for TeX as an Old Program?

Jean-Michel Hufflen

GuIT

31 October 2020
Teaching $\LaTeX$

$\TeX$ & $\LaTeX$, Yesterday and Today

Teaching Computer Science Students

More Ready for Next Time

Theoretically... Practically...

Conclusion
Teaching \LaTeX

Separation of *form* and *substance*.
Separation of *form* and *substance*. Typesetting system suitable for *large* documents.
Teaching \LaTeX

Separation of *form* and *substance*. Typesetting system suitable for *large* documents. *Resubmitting* an article.
Separation of *form* and *substance*. 
Typesetting system suitable for *large* documents. 
*Resubmitting* an article. 
*Markup* language.
Teaching \LaTeX

Separation of \textit{form} and \textit{substance}.

Typesetting system suitable for \textit{large} documents.

\textit{Resubmitting} an article.

\textit{Markup} language.

Included into some \textit{curricula} $\leftarrow$ GUIT 2019.
TEX, the Program — \LaTeX, the Format

- TEX’s 1st version $\leftarrow$ 1978.
TEX, the Program — \LaTeX, the Format

- \TeX’s 1st version $\leftrightarrow$ 1978.
- \LaTeX 2$\varepsilon$ $\leftrightarrow$ 1994.
TEX, the Program — \LaTeX, the Format

- \TeX{}'s 1st version ← 1978.
- \LaTeX{} 2\varepsilon ← 1994.
- \LaTeX{} 3 → intended to replace \LaTeX{} 2\varepsilon.
TEX, the Program — LaTeX, the Format

- TEX’s 1st version ← 1978.
- LaTeX $2\varepsilon$ ← 1994.
- LaTeX 3 → intended to replace LaTeX $2\varepsilon$.
- Another format ← ConTeXt (come out ca 1990).
Good

Still widespread $\iff$ very long lifetime.
Still widespread \iff very long lifetime.
Has incorporated modern requirements, e.g.:
Good

Still widespread ⇐ very long lifetime. Has incorporated modern requirements, e.g.:

▶ i18n,
Still widespread ⇐ very long lifetime. Has incorporated modern requirements, e.g.:

- i18n,
- new schemes for font management.
Still widespread ← very long lifetime. Has incorporated modern requirements, e.g.:

- i18n,
- new schemes for font management.

Much synergy among users, especially if you are interested in writing new commands.
Still widespread $\iff$ very long lifetime.
Has incorporated modern requirements, e.g.:

- i18n,
- new schemes for font management.

Much synergy among users, especially if you are interested in writing new commands.

\LaTeX\ 3 $\implies$ better for such a task,
Still widespread \iff very long lifetime. Has incorporated modern requirements, e.g.:

- i18n,
- new schemes for font management.

Much synergy among users, especially if you are interested in writing new commands.

\LaTeX{} 3 \implies better for such a task, but not finished yet.
Less Good

\TeX\’s implementation:

\begin{itemize}
  \item \TeX\’s implementation:
  \item \TeX\’s implementation:
  \item \TeX\’s implementation:
  \item \TeX\’s implementation:
  \item \TeX\’s implementation:
\end{itemize}
Less Good

TEX’s implementation:

▶ very old-fashioned language (based on Pascal)
Less Good

\textsc{\textsc{T}e\textsc{X}}'s implementation:

- very old-fashioned language (based on Pascal)
- monolithic program
Less Good

\TeX’s implementation:
▶ very old-fashioned language (based on Pascal)
▶ monolithic program
▶ no one but D. Knuth can change it!
Less Good

\TeX’s implementation:

- very old-fashioned language (based on Pascal)
- monolithic program
- no one but D. Knuth can change it!

(\LaTeX)\TeX’s commands implemented by *macros*:
Less Good

\( \text{T}\text{E}X \)'s implementation:

- very old-fashioned language (based on Pascal)
- monolithic program
- no one but D. Knuth can change it!

(\( \text{L}\text{A}\text{E}X \)'s commands implemented by _macros_:

\[
\text{f}(2019 + 1) \quad \text{m}(2019 + 1)
\]
Computer Science is... 

... *a science!*
Computer Science is... 

... a science! \iff Techniquest and methods, but also some History.
Computer Science is... 

... a science! \leftarrow Techniques and methods, but also some History.

But many students do not perceive this point.
Parsing

‘Modern’ languages $\leftarrow$ reserved words.
Parsing

‘Modern’ languages $\leftrightarrow$ reserved words.
Lexical + syntactical analysis
Parsing

‘Modern’ languages $\leftrightarrow$ reserved words.
Lexical + syntactical analysis $\implies$ result in *trees*. 
‘Modern’ languages \textless{} reserved words.  
Lexical + syntactical analysis \rightarrow result in trees.
In \TeX

No reserved keyword in the present sense.
In \TeX

No reserved keyword in the present sense.
One analyser:
No reserved keyword in the present sense.
One analyser:

Typing \texttt{\frac{1}{2}} with \LaTeX\ is easy!
In TeX

No reserved keyword in the present sense.
One analyser:

Typing \frac{1}{2} with \LaTeX\ is easy!

Dynamic search for ‘\fi’ associated with ‘\if...’ (show).
Efficiency questions

\[ NT \text{S vs } \LaTeX. \]
Efficiency questions

$\LaTeX$ vs $\TeX$.

*short* commands vs *long* ones.
One run... and after that

Cross references (compare with Con\TeXt’s texexec command).
One run... and after that

Cross references (compare with ConTeXt’s texexec command).
Overfull boxes caused by wrong hyphenation.
One run... and after that

Cross references (compare with ConTExt’s texexec command).
Overfull boxes caused by wrong hyphenation.
Marginal notes.
Cross references (compare with Con\TeXt’s texexec command).
Overfull boxes caused by wrong hyphenation.
Marginal notes.
Working on final and irrevocable versions \iffalse commands such as \sloppy, \newpage, and some commands originating from the microtype package.
In theory

\LaTeX’s end-users should not use constructs from *Plain \TeX*. 
In theory

\LaTeX’s end-users should not use constructs from *Plain \TeX*. \texttt{\newcommand} instead of \texttt{\def}, ifthen package, etc.
\LaTeX’s end-users should not use constructs from *Plain \TeX.*  
`\newcommand` instead of `\def`, `ifthen` package, etc.

\LaTeX 3
In practice

\LaTeX’s services are added to \TeX’s, without hiding the latter.
\LaTeX’s services are added to \TeX’s, without hiding the latter. Example $\iff$ conventions for the \texttt{\input} command.
In practice

\LaTeX’s services are added to \TeX’s, without hiding the latter. Example $\Leftarrow$ conventions for the `\input` command. ‘$$\ldots\$$’ vs ‘\[\ldots\]\’. 
In practice

\LaTeX\’s services are added to \TeX\’s, without hiding the latter. Example ⇐ conventions for the \texttt{\input} command. ‘$$\ldots$$’ vs ‘\[\ldots\]’.

Some commands—e.g., \texttt{\xspace}—are not guaranteed 100%.
What is sure

In my university, many students in Computer Science typeset documents using \LaTeX, even if they do not have to.
In my university, many students in Computer Science typeset documents using \LaTeX, even if they do not have to. \LaTeX’s language is a good example for a *simple* and *specialised* language for *simple* commands.
In my university, many students in Computer Science typeset documents using \LaTeX, even if they do not have to. \LaTeX’s language is a good example for a *simple* and *specialised* language for *simple* commands. *Alternative* conventions, even if they are quite obsolete.
Ending

We can teach \LaTeX as an unrivalled type setting system whereas we can express that \TeX is a kind of *legacy program*. It should be able to provide new services because of some powerful features such as \LaTeXe and \LaTeX 3.
Ending

We can teach \LaTeX as an unrivalled type setting system whereas we can express that \TeX is a kind of legacy program. It should be able to provide new services because of some powerful features such as Lua\TeX and \LaTeX 3. Its qualities supersede its defects. The latter are in connection with implementation \implies Computer Science students. They can learn from these weaknesses.