siunitx: Past, present and future

Joseph Wright

joseph.wright@morningstar2.co.uk
An appeal for help

From: "Stefan Pinnow" <stefan...@tu-clausthal.de>
Newsgroups: comp.text.tex
Subject: SIunits bug
Date: Tue, 20 Nov 2007 21:53:03 +0100

Hello NG,
I want to report that \reciprocal, \rpsquare, \rpcubic, etc. output is written as "-1" instead of a $-1$, when the package option "textstyle" is used.
I tried to contact Mr. Heldoorn, but he didn’t answer until now.
Does anyone have an idea what to do?

Here the minimal example:
---
\documentclass{article}
  \usepackage[textstyle]{SIunits}
% \usepackage{SIunits}
\begin{document}
  \reciprocal\second, \rpsquare\metre, \rpcubic\metre or
  $\rpcubic\metre$ instead of $\metre^{-1}$
\end{document}
---

Best regards,
Stefan Pinnow

https://groups.google.com/forum/#!searchin/comp.text.tex/SIunits$20bug|sort:
date/comp.text.tex/CxWCaMmNCYc/Gkp2707muyEJ
Hello all,

As some of you may have noticed, following a recent bug report concerning the SIunits package, I have taken over as the package maintainer. I have uploaded a bug fix for the specific issue to CTAN, and so hopefully it will appear within a day or two.

It has been suggested by the maintainer of the SIconst package that integration of the two be would worth considering. Other suggestions have also been made in the newsgroup and by private mail. I am therefore planning to review the existing situation and see what improvements are needed/desirable. As well as SIunits and SIconst, I am going to look at numprint, units, unitsdef and hepunits for inspiration/points to consider/etc.

So far, I have some outline ideas, for example:

...
Early testing

\ProvidesPackage{si}%
[2008/02/18 v0.6
   A comprehensive (SI) units package]
First release

From: CTAN Announcements <ctan...@dante.de>
Subject: new CTAN package: siunitx
Date: Wed, 16 Apr 2008 20:23:06 +0100

From my upload daemon:

> Name of contribution: siunitx
> Author’s name: Joseph Wright
> Location on CTAN: /macros/latex/exptl/siunitx
> Summary description: A comprehensive (SI) units package
> License type: lppl

https://groups.google.com/forum/#!searchin/comp.text.tex/SIunits$20bug|sort:
date/comp.text.tex/PXw8H08GOHI/92fd8zehsrgkJ
Core features

- Parsing and formatting units
Core features

- Parsing and formatting units
- A single key-value interface
Core features

- Parsing and formatting units
- A single key–value interface
- Automatic number formatting
Core features

– Parsing and formatting units
– A single key–value interface
– Automatic number formatting
– Tabular alignment of numbers
From version 1 to version 2

\ProvidesPackage{siunitx}  
[2010/02/22 v1.4c  
  A comprehensive (SI) units package]
From version 1 to version 2

\ProvidesPackage{siunitx}
[2010/02/22 v1.4c
  A comprehensive (SI) units package]

– Internals other than unit parsing
taken from existing packages
– Sub-optimal key–value choices
– Essentially no internal API
– Poor self-coded loops
– …
Enter expl3

It’s all Will’s fault!
– Initial plan was to rewrite using ‘classical’ approaches
Enter expl3

- Initial plan was to rewrite using ‘classical’ approaches
- Amount of library code needed was *significant*
Enter expl3

- Initial plan was to rewrite using ‘classical’ approaches
- Amount of library code needed was significant
- For key–value support, wrote keys3, now l3keys
Enter expl3

- Initial plan was to rewrite using ‘classical’ approaches
- Amount of library code needed was significant
- For key–value support, wrote keys3, now l3keys
- New internal API approaches
Enter expl3

- Initial plan was to rewrite using ‘classical’ approaches
- Amount of library code needed was *significant*
- For key–value support, wrote keys3, now l3keys
- New internal API approaches
- Improved performance
From version 2 to version 3

\ProvidesExplPackage {siunitx} {2018/05/17} {2.7s}
 {A comprehensive (SI) units package}
From version 2 to version 3

\ProvidesExplPackage {siunitx} {2018/05/17} {2.7s}
   {A comprehensive (SI) units package}

- Assumptions about fonts: OpenType, etc.
- Development of expl3 ideas: code-level API
- Internals still too messy
- Testing purely the PDF documentation
- Monolithic source
- Still too slow
Fonts

Up to version 2

1. Detect current font type
2. \text (an \mbox)
3. \ensuremath or similar
4. Perhaps \text (again!)
5. Font command,
   e.g. \mathrm or \rmfamily
Fonts

Up to version 2

1. Detect current font type
2. \text (an \mbox)
3. \ensuremath or similar
4. Perhaps \text (again!)
5. Font command,
   e.g. \mathrm or \rmfamily

The new approach

1. Set any aspects \textit{that are needed}
2. Only use an \mbox if math version has to be altered
An example

Input:

\texttt{\textbackslash siunitx\_unit\_format:nN}
\{ \texttt{joule} / \texttt{per} \texttt{mole} \}
\texttt{l\_tmpa\_tl}
\texttt{tl\_show:N l\_tmpa\_tl}

Output:

\texttt{\textgreater l\_tmpa\_tl=\texttt{mathrm} \{J\}, \texttt{mathrm} \{mol\}^{-1}}
siunitx v3-alpha

Working

– Core functionality:
  – Unit parsing and formatting
  – Real number formatting
  – Tabular columns
– Existing API: \num, \SI, \si, S-column
– New (experimental) document API: \unit, \qty

To do

– Multi-part numbers
– Ranges and lists
– Mapping between v2 and v3
– Various compatibility settings
Thanks

– Stefan Pinnow
– Danie Els
– Marcel Heldoorn
– Harald Harders
– Eckhart Guthöhrlein

– David Carlisle
– Will Robertson
– Till Tantau
– Enrico Gregorio