

The `hepunits` L^AT_EX package

Andy Buckley, `andy@insectnation.org`

September 27, 2007

Abstract

The `hepunits` package extends the existing (and excellent) `SIunits` package to support units commonly used in high-energy physics. HEP uses a rather specialised set of units to describe measurements of energies, masses, momenta, reaction cross-sections, luminosities and so-on. Using this package will provide particle physicists with a consistent and accurate way to refer to dimensionful HEP quantities.

1 Recommended usage

The basic usage mode for `hepunits` is to place

```
\usepackage{hepunits}
```

in the preamble of your document. `hepunits` also supports the `SIunits` optional arguments (`thickspace`, `amssymb` and so-on), which are passed on directly to the `SIunits` package. By default the `mediumspace`, `thickqspace`, `squaren` and `textstyle` options are passed, but these can be overridden. `amssymb` and `squaren` are considered to be mutually exclusive options, but you can choose to pass neither option to `SIunits` by using the `hepunits noamssquareissue` option. Additionally, the `hepunits notextstyle` option can be used to turn the `SIunits textstyle` off. On the whole, though, you should be able to use `hepunits` with no options in most circumstances and are likely only to need them if you want access to the binary or derived units (using `binary` and `derived/derivedinbase` respectively).

Finally, the only `hepunits`-specific option is `noprefixcmds`. This is discussed at the end of the document and is probably only useful for macro language pedants!¹

¹No offence intended to macro language pedants, of course. . .

2 Requirements

`hepunits` requires the `SIunits`, `xspace` and `amsmath` packages to be installed as part of your \TeX distribution. I don't know of any distributions for which this isn't the case, so chances are you're safe to just install `hepunits` and use it right away!

3 Provided units

The HEP units provided by `hepunits` are listed in Tables 1 and 2 below. All the example outputs have been produced with a command like `\unit{1.0}{\langle unit \rangle}` where $\langle unit \rangle$ is one of the unit commands listed in the tables.

Unit command	Example
Lengths	
<code>\nm</code>	1.0 nm
<code>\micron</code>	1.0 μm
<code>\mm</code>	1.0 mm
<code>\cm</code>	1.0 cm
Times	
<code>\ns</code>	1.0 ns
<code>\ps</code>	1.0 ps
<code>\fs</code>	1.0 fs
<code>\as</code>	1.0 as
Rates	
<code>\mHz</code>	1.0 mHz
<code>\Hz</code>	1.0 Hz
<code>\kHz</code>	1.0 kHz
<code>\MHz</code>	1.0 MHz
<code>\GHz</code>	1.0 GHz
<code>\THz</code>	1.0 THz
Misc.	
<code>\mrad</code>	1.0 mrad
<code>\gauss</code>	1.0 G

Table 1: List of non-HEP specific units provided by `hepunits`

Note that a lot of these units have, for convenience, been provided as explicit commands with various SI prefixes, rather than just defining the base unit and using the `SIunits` prescription for the prefixes. Let's give a demo in case you

Unit command	Example
Luminosities	
<code>\invcmsqpersecond</code>	1.0 cm ⁻² s ⁻¹
<code>\invcmsqpersec</code>	1.0 cm ⁻² s ⁻¹
<code>\lumiunits</code>	1.0 cm ⁻² s ⁻¹
Cross-sections	
<code>\barn</code>	1.0 b
<code>\invbarn</code>	1.0 b ⁻¹
<code>\nanobarn</code>	1.0 nb
<code>\invnanobarn / \invnb</code>	1.0 nb ⁻¹
<code>\picobarn</code>	1.0 pb
<code>\invpicobarn / \invpb</code>	1.0 pb ⁻¹
<code>\femtobarn</code>	1.0 fb
<code>\invfemtobarn / \invfb</code>	1.0 fb ⁻¹
<code>\attobarn</code>	1.0 ab
<code>\invattobarn / \invab</code>	1.0 ab ⁻¹

Table 2: List of HEP-specific units provided by `hepunits`

don't know what I'm waffling about...the “usual” `SUunits` way of doing things is like this: `\unit{1.0}{\mega\evoverc}`. This produces “1.0 MeV/*c*” just like `\unit{1.0}{\MeVoverc}` would do.

I've chosen to provide the explicit prefixed commands for convenience: choose your own favourite way (the same applies even more so for most of the non-HEP units). If you are bothered about the explicitly prefixed commands clogging up the `LATEX` macro namespace then pass the `noprefixcmds` option to `hepunits` and the offending commands won't be defined at all. This will make life awkward when it comes to inverse cross-sections as used for integrated luminosities, but with suitable use of `\invbarn` I'm sure you can make do.

4 Summary

`hepunits` is a handy package for particle physicists who'd like their units to look right, with upright μ s and properly italicised *cs* in the appropriate places. Fortunately most of the work has already been done by the marvellous `SIunits` package and I've just provided a few more commands and an option passing wrapper on to that excellent piece of work.

If you have any comments, criticism, huge cash donations etc., then please do

Unit command	Example
eV-based units	
<code>\eV</code>	1.0 eV
<code>\inveV</code>	1.0 eV ⁻¹
<code>\eVoverc</code>	1.0 eV/ <i>c</i>
<code>\eVovercsq</code>	1.0 eV/ <i>c</i> ²
<code>\meV</code>	1.0 meV
<code>\keV</code>	1.0 keV
<code>\MeV</code>	1.0 MeV
<code>\GeV</code>	1.0 GeV
<code>\TeV</code>	1.0 TeV
<code>\minveV</code>	1.0 meV ⁻¹
<code>\kinveV</code>	1.0 keV ⁻¹
<code>\MinveV</code>	1.0 MeV ⁻¹
<code>\GinveV</code>	1.0 GeV ⁻¹
<code>\TinveV</code>	1.0 TeV ⁻¹
<code>\meVoverc</code>	1.0 meV/ <i>c</i>
<code>\keVoverc</code>	1.0 keV/ <i>c</i>
<code>\MeVoverc</code>	1.0 MeV/ <i>c</i>
<code>\GeVoverc</code>	1.0 GeV/ <i>c</i>
<code>\TeVoverc</code>	1.0 TeV/ <i>c</i>
<code>\meVovercsq</code>	1.0 meV/ <i>c</i> ²
<code>\keVovercsq</code>	1.0 keV/ <i>c</i> ²
<code>\MeVovercsq</code>	1.0 MeV/ <i>c</i> ²
<code>\GeVovercsq</code>	1.0 GeV/ <i>c</i> ²
<code>\TeVovercsq</code>	1.0 TeV/ <i>c</i> ²

Table 2: List of HEP-specific units provided by `hepunits` (cont.)

send them my way. Email to andy@insectnation.org is preferred, but if you can find a way to get your message to me by carrier pigeon I'll be very impressed.