

annotate-equations.sty

ST John

<https://github.com/st--/annotate-equations>

1 Introduction

This package is there to make it easier to make annotated equations in L^AT_EX, such as in this example:

The image shows the Schrödinger equation $i\hbar \frac{\partial}{\partial t} \Psi(x, t) = \hat{H} \Psi(x, t)$. Annotations include: a line above \hbar pointing to $\hbar = \frac{h}{2\pi}$, reduced Planck constant; a red arrow pointing to \hat{H} labeled Hamilton operator; and a blue arrow pointing to $\Psi(x, t)$ labeled Wave function.

```
\vspace{4em}
\renewcommand{\eqnhighlightheight}{\vphantom{\hat{H}}\mathstrut}
\begin{equation*}
  i \tikzmarknode{hbar}{\mathstrut\hbar} \frac{\partial}{\partial t}
  \eqnmarkbox[blue]{Psi1}{\Psi(x, t)} = \eqnmark[red]{Hhat}{\hat{H}}
  \eqnmarkbox[blue]{Psi2}{\Psi(x, t)}
\end{equation*}
\annotate[yshift=3em]{above}{hbar}{\hbar = \frac{h}{2\pi}$, reduced Planck constant}
\annotate[yshift=1em]{above}{Hhat}{Hamilton operator}
\annotatetwo[yshift=-1em]{below}{Psi1}{Psi2}{Wave function}
\vspace{1em}
```

There is still a bit of manual tweaking required (such as adding vertical space before/after the equation), but hopefully this package will already make it a bit more inviting to annotate your equations!

Note that this package relies on TikZ's `remember picture` option and therefore you have to compile your L^AT_EX document at least twice to get everything in the right place (or just use `latexmk!`).

2 Marking annotation targets within your equation

Use `\eqnmarkbox[<color>]{<node name>}{<equation term(s)>}` or `\eqnmark[<color>]{<node name>}{<equation term(s)>}` to define the target of an annotation within your equation. `\eqnmarkbox` adds background shading, whereas `\eqnmark` changes the text color. (You can also use `\tikzmarknode{<node name>}{<equation term(s)>}`, but this is likely to end up with the arrow tip too close to the target, so you may want to also pass the `outer ysep` option, see section 6.3.)

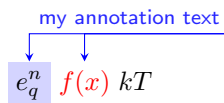
The image shows the equation $e_q^n f(x) kT$. Annotations include: a blue box around e_q^n labeled node1; a red box around $f(x)$ labeled node2; and a blue box around kT labeled node3.

```
\begin{equation*}
  \eqnmarkbox[blue]{node1}{e_q^n}
  \eqnmark[red]{node2}{f(x)}
  \tikzmarknode{node3}{kT}
\end{equation*}
```

3 Simple annotations

Once you have defined nodes within your equations, you can annotate them using `\annotate[<tikz options>]{<annotate keys>}{<node name[,...]>}{<annotation text>}`. `<tikz options>` is passed through to

the options for the TikZ node defining the annotation; its most important use is to set the `yshift`. For `\annotate keys`, see section 3.1. `\node name` is the same name you used to mark the node within the equation, e.g. using `\eqnmarkbox`. `\annotation text` is the text of the annotation itself.



```
\begin{equation*}
  \eqnmarkbox[blue]{node1}{e_q^n}
  \eqnmark[red]{node2}{f(x)}
  \tikzmarknode{node3}{kT}
\end{equation*}
\annotate[yshift=1em]{}{node1,node2}{my annotation text}
```

You generally need to manually adjust the `yshift` to move the annotations to an appropriate distance above (or negative values for below) the equation. If you want an annotation below the equation, with negative `yshift`, remember to also pass the `below` option (see section 3.1). (You can also adjust `xshift` if needed, also positive or negative.)

The annotation picks the same text color as given to `\eqnmarkbox` or `\eqnmark`, but you can also override it using `color` option.

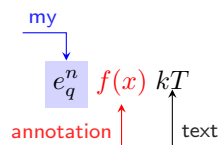
One annotation can point to multiple targets, and multiple annotations can point to the same target.

3.1 Annotation options

`\annotate keys` can be empty, or contain one or more of:

- `above` (default) or `below`,
- `right` (default) or `left`,
- `label above` (default) or `label below`.

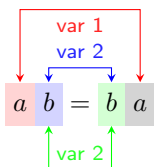
Note: currently only works for `\annotatetwo` (section 4).



```
\begin{equation*}
  \eqnmarkbox[blue]{node1}{e_q^n}
  \eqnmark[red]{node2}{f(x)}
  \tikzmarknode{node3}{kT}
\end{equation*}
\annotate[yshift=1em]{left}{node1}{my}
\annotate[yshift=-0.5em]{below,left}{node2}{annotation}
\annotate[yshift=-1em]{below}{node3}{text}
```

4 Double annotations

`\annotatetwo[tikz options]{annotate keys}{first node name}{second node name}{annotation text}`. `\tikz options` and `\annotate keys` are as described above in sections 3 and 3.1. Note that `\annotate keys` `left/right` have no effect in `\annotatetwo`.



```
\begin{equation*}
  \eqnmarkbox[red]{a1}{a} \eqnmarkbox[blue]{b1}{b} =
  \eqnmarkbox[green]{b2}{b} \eqnmarkbox{a2}{a}
\end{equation*}
\annotatetwo[yshift=1.5em]{above, label below}{a1}{a2}{var 1}
\annotatetwo[yshift=0.5em]{above}{b1}{b2}{var 2}
\annotatetwo[yshift=-0.5em]{below}{b2}{b1}{var 2}
```

Color is picked from the first of the two nodes.

5 Package options

5.1 Size of highlight: shrink to content or always full height

`\eqnhighlightheight` is inserted into every `\eqnhighlight`, `\eqncolor`, `\eqnmark`, and `\eqnmarkbox` and by redefining it you can specify the minimum height for the corresponding box:

\hbar q

```
\renewcommand{\eqnhighlightheight}{} % package default

\begin{equation*}
  \eqnmarkbox[red]{\hbar}{\hbar} \eqnmarkbox[blue]{q}{q}
\end{equation*}
```

\hbar q

```
\renewcommand{\eqnhighlightheight}{\mathstrut} % 0-width "constant" height

\begin{equation*}
  \eqnmarkbox[red]{\hbar}{\hbar} \eqnmarkbox[blue]{q}{q}
\end{equation*}
```

`\eqnhighlightheight` is used in math mode.

Note that in some cases `\mathstrut` might not be enough, as in the introductory example:

\hat{H} Ψ

```
\renewcommand{\eqnhighlightheight}{\mathstrut} % 0-width "constant" height

\begin{equation*}
  \eqnmarkbox[red]{\hat{H}}{\hat{H}} \eqnmarkbox[blue]{\Psi}{\Psi}
\end{equation*}
```

You can create custom 0-width characters using `\vphantom`:

\hat{H} Ψ

```
\renewcommand{\eqnhighlightheight}{\vphantom{\hat{H}}\mathstrut} % custom
  0-width height

\begin{equation*}
  \eqnmarkbox[red]{\hat{H}}{\hat{H}} \eqnmarkbox[blue]{\Psi}{\Psi}
\end{equation*}
```

(It looks more balanced if you still include the `\mathstrut`.)

5.2 Amount of shading of mark highlight

`\eqnhighlightshade` defines the percentage of the specified color to take:

\hbar q

```
\renewcommand{\eqnhighlightshade}{17} % package default

\begin{equation*}
  \eqnmarkbox[red]{\hbar}{\hbar} \eqnmarkbox[blue]{q}{q}
\end{equation*}
```

By redefining this command, you can change the “alpha” value of the highlight:

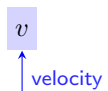
\hbar q

```
\renewcommand{\eqnhighlightshade}{47} % 0 is white, 100 is solid color

\begin{equation*}
  \eqnmarkbox[red]{\hbar}{\hbar} \eqnmarkbox[blue]{q}{q}
\end{equation*}
```

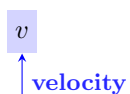
5.3 Default formatting of annotation labels

`\eqnannotationfont` sets the font field of the TikZ annotation label and can be re-set to change its formatting:



```
\renewcommand{\eqnannotationfont}{\sffamily\footnotesize} % package default

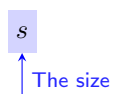
\begin{equation*}
  \eqnmarkbox[blue]{v}{v}
\end{equation*}
\annotate[yshift=-0.5em]{below}{v}{velocity}
\vspace{1em}
```



```
\renewcommand{\eqnannotationfont}{\bfseries\small}

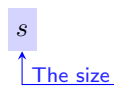
\begin{equation*}
  \eqnmarkbox[blue]{v}{v}
\end{equation*}
\annotate[yshift=-0.5em]{below}{v}{velocity}
\vspace{1em}
```

`\eqnannotationstrut` is defined to be a strut (zero-width height) to provide minimum distance between the text and the corresponding arrow line. By default it is `\strut`, which has a similar effect to `\mathstrut` in `\eqnhighlightheight`.



```
\renewcommand{\eqnannotationstrut}{\strut} % package default

\begin{equation*}
  \eqnmarkbox[blue]{size}{s}
\end{equation*}
\annotate[yshift=-0.5em]{below}{size}{The size}
\vspace{1em}
```



```
\renewcommand{\eqnannotationstrut}{}

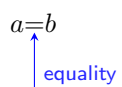
\begin{equation*}
  \eqnmarkbox[blue]{size}{s}
\end{equation*}
\annotate[yshift=-0.5em]{below}{size}{The size}
\vspace{1em}
```

6 Recommendations, tips & tricks

6.1 Use `\colorlet` for consistent, easily changeable colors

6.2 Relations such as “=”

Wrapping a mathematical relation symbol such as `=` in, for example, `\tikzmarknode`, breaks how \TeX determines spacing in equations:



```
\[
  a \tikzmarknode{node1}{=} b
\]
\annotate[yshift=-1em]{below}{node1}{equality}
```

This can be fixed by wrapping the `\tikzmarknode` in `\mathrel`:

$$a = b$$

equality

```
\[
a \mathrel{\tikzmarknode{node1}{=}} b
\]
\annotate[yshift=-1em]{below}{node1}{equality}
```

6.3 Extra spacing between `\tikzmarknode` and arrow

If you want more space between arrow tip and annotated term, you can pass the outer `ysep` option to `\tikzmarknode`:

$$a = b$$

equality

```
\[
a \mathrel{\tikzmarknode[outer ysep=5pt]{node1}{=}} b
\]
\annotate[yshift=-1em]{below}{node1}{equality}
```

7 Known issues

- `label above/label below` is not implemented for `\annotate`.
- Annotations of mathematical relations require some manual patching to get the correct surrounding spacing (see section 6.2).