

# annotate-equations.sty, v.0.2.2

ST John

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

This package is there to make it easier to make annotated equations in L<sup>A</sup>T<sub>E</sub>X, such as in this example:

The image shows the Schrödinger equation  $i\hbar \frac{\partial}{\partial t} \Psi(x, t) = \hat{H} \Psi(x, t)$  with three annotations. A black arrow points from the text " $\hbar = \frac{h}{2\pi}$ , reduced Planck constant" to the  $\hbar$  symbol. A red arrow points from the text "Hamilton operator" to the  $\hat{H}$  symbol. A blue arrow points from the text "Wave function" to the  $\Psi(x, t)$  term on the right side of the equation. The terms  $\Psi(x, t)$  and  $\hat{H}$  are highlighted with blue and red boxes respectively.

```
\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}{\mathstrut\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<sup>A</sup>T<sub>E</sub>X document at least twice to get everything in the right place (or just use `latexmk`!).

## Contents

<b>1</b>	<b>Marking annotation targets within your equation</b>	<b>2</b>
<b>2</b>	<b>Simple annotations</b>	<b>2</b>
2.1	Annotation options . . . . .	2
<b>3</b>	<b>Double annotations</b>	<b>3</b>
<b>4</b>	<b>Package options</b>	<b>3</b>
4.1	Size of highlight: shrink to content or always full height . . . . .	3
4.2	Amount of shading of mark highlight . . . . .	4
4.3	Default formatting of annotation labels . . . . .	4
4.4	Customize style . . . . .	5
<b>5</b>	<b>Recommendations, tips &amp; tricks</b>	<b>6</b>
5.1	Use <code>\colorlet</code> for consistent, easily changeable colors . . . . .	6
5.2	Relations such as “=” . . . . .	6
5.3	Extra spacing between <code>\tikzmarknode</code> and arrow . . . . .	6
<b>6</b>	<b>Known issues</b>	<b>6</b>

## 1 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 5.3.)

$$e_q^n \quad f(x) \quad kT$$

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

## 2 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 2.1. `⟨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.

$$e_q^n \quad f(x) \quad kT$$

my annotation text

```
\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 2.1). (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.

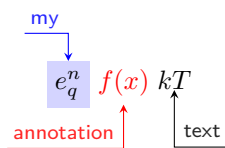
One annotation can point to multiple targets, and multiple annotations can point to the same target.

### 2.1 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`.

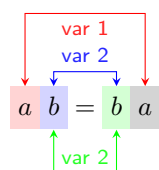
Note: currently only works for `\annotatetwo` (section 3).



```
\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, label below}{node3}{text}
```

### 3 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 2 and 2.1. 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.

## 4 Package options

### 4.1 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:



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

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



```
\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:



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

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

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

```
\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`.)

## 4.2 Amount of shading of mark highlight

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

```
\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:

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

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

## 4.3 Default formatting of annotation labels

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

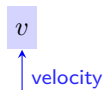
```
\renewcommand{\eqnannotationfont}{\sffamily\footnotesize} % package
                                default

\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}
```

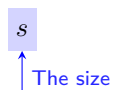
Alternatively, you can also change the style of `annotate` equations/text:



```
\tikzset{annotate equations/text/.style={font=\bfseries\small}}

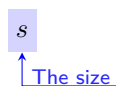
\begin{equation*}
  \eqnmarkbox[blue]{v}{v}
\end{equation*}
\annotate[yshift=-0.5em]{below}{v}{velocity}
\vspace{1em}
```

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

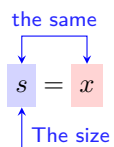


```
\renewcommand{\eqnannotationstrut}{}

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

## 4.4 Customize style

You can change the style of the annotation arrow line by setting the style of `annotate equations/arrow`:

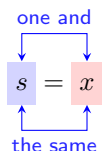


```
\tikzset{annotate equations/arrow/.style={color=ForestGreen, >=latex',
  very thick, dashed}}

\begin{equation*}
  \eqnmarkbox[blue]{size}{s} = \eqnmarkbox[red]{other}{x}
\end{equation*}
\annotate[yshift=-0.5em]{below}{size}{The size}
\annotatetwo[yshift=1em]{above}{size}{other}{the same}
```

Note that it applies to all `\annotate` and `\annotatetwo` arrows within the scope.

You can also use this to change the arrow direction:



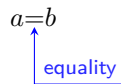
```
\begin{equation*}
  \eqnmarkbox[blue]{size}{s} = \eqnmarkbox[red]{other}{x}
\end{equation*}
\tikzset{annotate equations/arrow/.style={->}}
\annotatetwo[yshift=1em]{above}{size}{other}{one and}
\tikzset{annotate equations/arrow/.style={<-}}
\annotatetwo[yshift=-1em]{below, label below}{size}{other}{the same} %
  note that the "direction" of the arrow is from first to second mark
```

## 5 Recommendations, tips & tricks

### 5.1 Use `\colorlet` for consistent, easily changeable colors

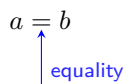
### 5.2 Relations such as “=”

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}
```

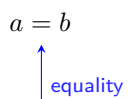
This can be fixed by wrapping the `\tikzmarknode` in `\mathrel`:



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

### 5.3 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`:



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

## 6 Known issues

- Annotations of mathematical relations require some manual patching to get the correct surrounding spacing (see section 5.2).

## 7 Backwards-incompatible changes

### v0.2.0

`\eqnannotationtext` removed

To make it easier to format multiline annotations, version 0.2.0 introduced the `\eqnannotationfont` and `\eqnannotationstrut` (zero-argument) commands (see section 4.3).

In exchange, the `\eqnannotationtext` (one-argument) command was removed. To upgrade, replace for example

```
\renewcommand{\eqnannotationtext}[1]{\sffamily\tiny#1\strut}
```

with

```
\renewcommand{\eqnannotationfont}{\sffamily\tiny}
\renewcommand{\eqnannotationstrut}{\strut}
```