The `physics-patch` package

Improved version of the physics package

Willie Shen (Willie169)

Version 2.9
Last update: May 21, 2025

Contents

1 Preface

Since version 2.0, the physics-patch package has evolved from merely patching the physics package to fully replacing it. While preserving the original goal—simplifying mathematical and physics typesetting for greater readability and efficiency—this package refines the design by addressing unintuitive behaviors, extending commands, and introducing additional macros. For instance, in the original package, parentheses and their contents after \dv{f}{x} are ignored. This package also extends commands for broader applicability such as enabling \xmat to support ellipses and introducing entirely new macros such as \omat.

2 Usage

2.1 Required packages

The physics-patch package requires amsmath, etoolbox, xcolor, xparse, and xstring package to work. If you are unsure whether you've had them installed, you can either install it again using your local package manager (comes with most distributions) or by visiting the CTAN online package database, or even just try to use physics-patch package without worrying about it. Many modern LaTeXcompilers will locate and offer to download missing required packages for you.

2.2 Using physics-patch in your LaTeX document

To use physics-patch in your LaTeX document, simply insert \usepackage{physics-patch} in the preamble of your document, before \begin{document} and after \documentclass{class}:

```
\documentclass{class}
...
\usepackage{physics-patch}
...
\begin{document}
content...
\end{document}
```

This package will silently override the commands that have been defined before this package is loaded. To use the original definition provided by physics, load physics before this package and use the nooverride option for this package (not recommended). nooverride falls back to the original behavior if physics has not been loaded.

This package pretends that physics is loaded so that this package won't be overriden if loading physics is called afterward and packages that depend on physics (e.g. siunitx) work correctly. To disable this, use the nopretend option (not recommended).

If siuitx is loaded before this package, this package will define \ITquantity and \ITqty with the integration of the revised definition of physics's \qty (in \PHquantity and \PHqty) and siuitx's \SI. You can optionally set siintegrate option to override \PTquantity and \PTqty with \ITqty (not recommended). siintegrate falls back to the original behavior if siunitx is not loaded.

2.3 Options

Options available in this package include

- `nooverride`: Not override macros in physics to patched ones (not recommended). Fall back to override if physics is not loaded.
- `override`: Override macros in physics to patched ones (default). This option can be set no matter whether physics is loaded.
- `nopretend`: Not pretend that physics package is loaded (not recommend).
- `pretend`: Pretend that `physics` is loaded so that this package won't be overriden if loading `physics` is called afterward and packages that depend on `physics` (e.g. `siunitx`) work correctly.
- `nosiintegrate`: Not override `\PTquantity` and `\PTqty` with `\ITqty`.
- `siintegrate`: Override `\PTquantity` and `\PTqty` with `\ITqty` (not recommended). Fall back to nosiintegrate if siunitx is not loaded.
- `nooriginaldiv`: Let `\div` be `\divergence` (not recommended).
- `originaldiv`: Let `\div` be division symbol (default).
- `notrig`: Not redefine trig function and operator.
- `trig`: Redefine trig function and operator (default).
- `italicdiff`: Italic differentials.
- `uprightdiff`: Upright differentials (default).
- `arrowdel`: Vector arrow `\nabla` symbol.
- `bolddel`: Vector bold `\nabla` symbol (default).
- `bolddot`: Vector bold dot product symbol.
- `plaindot`: Vector plain dot product symbol (default).
- `noshorttextgreek`: Not define shorthands for text Greek alphabet (default).
- `shorttextgreek`: Define shorthands for text Greek alphabet.
- `noshortvargreek`: Not define shorthands for variant Greek alphabet (default).
- `shortvargreek`: Define shorthands for variant Greek alphabet.
- `noshortupgreek`: Not define shorthands for upright Greek alphabet (default).
- `shortupgreek`: Define shorthands for upright Greek alphabet.
- `noshortupvargreek`: Not define shorthands for upright variant Greek alphabet (default).
- `shortupvargreek`: Define shorthands for upright variant Greek alphabet.
- `noshortboldgreek`: Not define shorthands for bold Greek alphabet (default).
- `shortboldgreek`: Define shorthands for bold Greek alphabet.
- `noshortgreek`: For backward compatibility. Do nothing.
- `shortgreek`: Define all shorthands for Greek alphabet.

- `noshortmathrm`: Not define shorthands for mathrm alphabet and chemical element symbols (default).
- `shortmathrm`: Define shorthands for mathrm alphabet and chemical element symbols.
- `noshorttext`: Not define shorthands for textnormal alphabet (default).
- `shorttext`: Define shorthands for textnormal alphabet.

To use all features of this package, load it with `\usepackage[override,pretend,nosiintegrate,origi`

If two opposite options which one of them with the name of the other prefixed with a no are loaded at the same time, the one without the no in the name will be used. If two opposite options which have the same suffix and different prefix are loaded at the same time, the default one will be used. If `shortgreek` option is used, `noshorttextgreek`, `noshortvargreek`, `noshortupgreek`, `noshortupvargreek`, and `noshortboldgreek` are ignored.

3 Communication Channels

- **Bug tracker:** https://github.com/Willie169/physics-patch/issues.
- **Announcements:** https://github.com/Willie169/physics-patch/releases.
- **Repository:** https://github.com/Willie169/physics-patch.

4 License and Credit

- This package is released under the **LaTeX Project Public License (LPPL) 1.3c.**

  See https://www.latex-project.org/lppl/lppl-1-3c for the details of that license.

- Many parts of this package are modified or copied from the `physics` package, created by **Sergio C. de la Barrera** and licenced under **LPPL 1.3**.

  See https://ctan.org/pkg/physics for the details of that package.

- Many parts of this package rely on the `amsmath` package, created by **The LaTeX Project Team** and licenced under **LPPL 1.3c**.

  See https://ctan.org/pkg/amsmath for the details of that package.

5 List of Commands

In the commands listed below, the left column is long-form names with non-default alternate names (if any), the middle column is default shorthand commands with detailed syntaxes and explanations.

If `nooverride` is not used or the `physics` package is not loaded before this package, the commands prefixed with `PT` will silently override those without `PT`.

5.1 Automatic bracing

`\PTquantity`, `\PHquantity` or `\PHqty`	`\PTqty(\typical)` → $(\blacksquare)$	automatic ( ) braces
	`\PTqty(\tall)` → $\left(\blacksquare\right)$	
	`\PTqty(\grande)` → $\left(\blacksquare\right)$	
	`\PTqty[\typical]` → $[\blacksquare]$	automatic [ ] braces
	`\PTqty|\typical|` → $\|\blacksquare\|$	automatic | | braces
	`\PTqty{\typical}` → $\{\blacksquare\}$	automatic { } braces
	`\PTqty\big{}` → $\big\{\big\}$	manual sizing (works with any of the above bracket types)
	`\PTqty\Big{}` → $\Big\{\Big\}$	
	`\PTqty\bigg{}` → $\bigg\{\bigg\}$	
	`\PTqty\Bigg{}` → $\Bigg\{\Bigg\}$	
	`\pqty{}` ↔ `\PTqty()`	alternative syntax; robust and more LaTeX-friendly
	`\bqty{}` ↔ `\PTqty[]`	
	`\vqty{}` ↔ `\PTqty||`	
	`\Bqty{}` ↔ `\PTqty{}`	
`\absolutevalue`	`\abs{a}` → $\lvert a\rvert$	automatic sizing; equivalent to `\PTqty|a|`
	`\abs\Big{a}` → $\Big\lvert a\Big\rvert$	inherits manual sizing syntax from `\PTqty`
	`\abs*{\grande}` → $\lvert\blacksquare\rvert$	star for no resize
`\norm`	`\norm{a}` → $\lVert a\rVert$	automatic sizing
	`\norm\Big{a}` → $\Big\lVert a\Big\rVert$	manual sizing
	`\norm*{\grande}` → $\lVert\blacksquare\rVert$	star for no resize
`\evaluated`	`\eval{x}_0^\infty` → $x\Big\rvert_0^\infty$	vertical bar for evaluation limits
	`\eval(x|_0^\infty` → $\left(x\right\rvert_0^\infty$	alternate form
	`\eval[x|_0^\infty` → $\left[x\right\rvert_0^\infty$	alternate form
	`\eval[\venti|_0^\infty` → $\left[\blacksquare\right\rvert_0^\infty$	automatic sizing
	`\eval*[\venti|_0^\infty` → $\left[\blacksquare\right\rvert_0^\infty$	star for no resize
`\order`	`\order{x^2}` → $\mathcal{O}\left(x^2\right)$	order symbol; automatic sizing and space handling
	`\order\Big{x^2}` → $\mathcal{O}\Big(x^2\Big)$	manual sizing
	`\order*{\grande}` → $\mathcal{O}(\blacksquare)$	star for no resize
`\commutator`	`\comm{A}{B}` → $[A, B]$	automatic sizing

	`\comm\Big{A}{B}` → $\Big[A, B\Big]$ `\comm*{A}{\grande}` → $[A, \grande]$	manual sizing star for no resize
`\anticommutator` or `\acommutator`	`\acomm{A}{B}` → $\{A, B\}$	same as `\poissonbracket`
`\poissonbracket`	`\pb{A}{B}` → $\{A, B\}$	same as `\anticommutator`

5.2 Vector notation

The default del symbol ∇ used in `physics-patch` vector notation can be switched to appear with an arrow $\vec{\nabla}$ by including the option `arrowdel` in the document preamble:

```
\usepackage[arrowdel]{physics-patch}
```

`\vectorbold`	`\vb{a}` → $\mathbf{a}$ `\vb*{a}`, `\vb*{\theta}` → $\boldsymbol{a}$, $\boldsymbol{\theta}$	upright/no Greek italic/Greek
`\vectorarrow`	`\va{a}` → $\vec{\mathbf{a}}$ `\va*{a}`, `\va*{\theta}` → $\vec{\boldsymbol{a}}$, $\vec{\boldsymbol{\theta}}$	upright/no Greek italic/Greek
`\vectorunit`	`\vu{a}` → $\hat{\mathbf{a}}$ `\vu*{a}`, `\vu*{\theta}` → $\hat{\boldsymbol{a}}$, $\hat{\boldsymbol{\theta}}$	upright/no Greek italic/Greek
`\dotproduct`	`\vdot` → $\cdot$ as in $\mathbf{a} \cdot \mathbf{b}$	note: `\dp` is a protected TEX primitive
`\crossproduct`	`\cross` → $\times$ as in $\mathbf{a} \times \mathbf{b}$ `\cp` → $\times$ as in $\mathbf{a} \times \mathbf{b}$	alternate name shorthand name

`\gradient`	`\grad` → ∇
	`\grad{\Psi}` → $\nabla\Psi$
	`\grad(\Psi+\tall)`
	→ $\nabla\left(\Psi + \tall\right)$
	`\grad[\Psi+\tall]`
	→ $\nabla\left[\Psi + \tall\right]$
		default mode
		long-form (like `\PTqty` but also handles spacing)

`\divisionsymbol`	`\divisionsymbol` → $\div$	

`\divergence`	`\divg` → $\nabla\cdot$
	`\divg{\vb{a}}` → $\nabla \cdot \mathbf{a}$
	`\divg(\vb{a}+\tall)`
	→ $\nabla \cdot \left(\mathbf{a} + \tall\right)$
	`\divg[\vb{a}+\tall]`
	→ $\nabla \cdot \left[\mathbf{a} + \tall\right]$
		note: if `nooriginaldiv` option is used, `\div` will be overriden as $\nabla\cdot$ too (not recommended)
		default mode
		long-form

`\curl`	`\curl` → $\nabla\times$	

	\curl{\vb{a}} → $\nabla \times \mathbf{a}$ \curl(\vb{a}+\tall) → $\nabla \times \left(\mathbf{a} + \blacksquare\right)$ \curl[\vb{a}+\tall] → $\nabla \times \left[\mathbf{a} + \blacksquare\right]$	default mode long-form
\laplacian	\laplacian → ∇^2 \laplacian{\Psi} → $\nabla^2 \Psi$ \laplacian(\Psi+\tall) → $\nabla^2\left(\Psi + \blacksquare\right)$ \laplacian[\Psi+\tall] → $\nabla^2\left[\Psi + \blacksquare\right]$	default mode long-form

5.3 Operators

The standard set of trig functions is redefined in physics-patch to provide automatic braces that behave like \PTqty(). In addition, an optional power argument is provided. This behavior can be switched off by including the option notrig in the preamble:

```
\usepackage[notrig]{physics-patch}
```

Example trig redefinitions:		
\sin	\sin(\grande) → $\sin\left(\blacksquare\right)$ \sin[2](x) → $\sin^2(x)$ \sin x → $\sin x$	automatic braces; old \sin renamed \sine optional power can still use without an argument

The full set of available trig functions in physics-patch includes:

```
\sin(x)   \sinh(x)   \arcsin(x)   \asin(x)
\cos(x)   \cosh(x)   \arccos(x)   \acos(x)
\tan(x)   \tanh(x)   \arctan(x)   \atan(x)
\csc(x)   \csch(x)   \arccsc(x)   \acsc(x)
\sec(x)   \sech(x)   \arcsec(x)   \asec(x)
\cot(x)   \coth(x)   \arccot(x)   \acot(x)
```

⇒

$$\begin{array}{cccc}
\sin(x) & \sinh(x) & \arcsin(x) & \asin(x) \\
\cos(x) & \cosh(x) & \arccos(x) & \acos(x) \\
\tan(x) & \tanh(x) & \arctan(x) & \atan(x) \\
\csc(x) & \csch(x) & \arccsc(x) & \acsc(x) \\
\sec(x) & \sech(x) & \arcsec(x) & \asec(x) \\
\cot(x) & \coth(x) & \arccot(x) & \acot(x)
\end{array}$$

The standard trig functions (plus a few that are missing in `amsmath`) are available without any automatic bracing under a new set of longer names:

`\sine`	`\hypsine`	`\arcsine`	`\asine`
`\cosine`	`\hypcosine`	`\arccosine`	`\acosine`
`\tangent`	`\hyptangent`	`\arctangent`	`\atangent`
`\cosecant`	`\hypcosecant`	`\arccosecant`	`\acosecant`
`\secant`	`\hypsecant`	`\arcsecant`	`\asecant`
`\cotangent`	`\hypcotangent`	`\arccotangent`	`\acotangent`

Similar behavior has also been extended to the following functions:

`\exp(\tall)`	$\exp\left(\blacksquare\right)$	old definitions ⇒	`\exponential`
`\log(\tall)`	$\log\left(\blacksquare\right)$		`\logarithm`
`\ln(\tall)`	$\ln\left(\blacksquare\right)$		`\naturallogarithm`
`\det(\tall)`	$\det\left(\blacksquare\right)$		`\determinant`
`\Pr(\tall)`	$\Pr\left(\blacksquare\right)$		`\Probability`
`\Arg(\tall)`	$\mathrm{Arg}\left(\blacksquare\right)$		`\Argument`
`\arg(\tall)`	$\arg\left(\blacksquare\right)$		`\argument`
`\Re(\tall)`	$\Re(\blacksquare)$		`\real`
`\Im(\tall)`	$\Im(\blacksquare)$		`\imaginary`

New operators:		
`\tr`	`\tr\rho` → $\mathrm{tr}\,\rho$ also `\tr(\tall)` → $\mathrm{tr}\left(\blacksquare\right)$	trace; same bracing as trig functions
`\Tr`	`\Tr\rho` → $\mathrm{Tr}\,\rho$	alternate
`\rank`	`\rank M` → $\mathrm{rank}\,\boldsymbol{M}$	matrix rank
`\erf`	`\erf(x)` → $\mathrm{erf}(x)$	error function
`\Res`	`\Res[f(z)]` → $\mathrm{Res}[f(z)]$	residue; same bracing as trig functions
`\acosh`	`\acosh(\pi)` → $\mathrm{acosh}(\pi)$	acosh
`\acsch`	`\acsch(\pi)` → $\mathrm{acsch}(\pi)$	acsch
`\arccosh`	`\arccosh(\pi)` → $\mathrm{arccosh}(\pi)$	arccosh
`\arccsch`	`\arccsch(\pi)` → $\mathrm{arccsch}(\pi)$	arccsch
`\arcsech`	`\arcsech(\pi)` → $\mathrm{arcsech}(\pi)$	arcsech
`\arcsinh`	`\arcsinh(\pi)` → $\mathrm{arcsinh}(\pi)$	arcsinh
`\arctanh`	`\arctanh(\pi)` → $\mathrm{arctanh}(\pi)$	arctanh
`\arctantwo`	`\arctantwo(\pi)` → $\mathrm{arctan2}(\pi)$	arctan2

\asech	\asech(\pi) → asech(π)	asech
\asinh	\asinh(\pi) → asinh(π)	asinh
\atanh	\atanh(\pi) → atanh(π)	atanh
\atantwo	\atantwo(\pi) → atan2(π)	atan2
\closure	\closure(A) →$\circlearrowleft$ (A)	closure
\col	\col(\mathbf{A}) → col($\mathbf{A}$)	column space
\Col	\Col(\mathbf{A}) → Col($\mathbf{A}$)	column space
\dim	\dim(V) → dim(V)	dimension
\distance	\distance(A,B) → distance(A, B)	lowercase distance
\Distance	\Distance(A,B) → Distance(A, B)	uppercase distance
\row	\row(\mathbf{A}) → row($\mathbf{A}$)	row space
\Row	\Row(\mathbf{A}) → Row($\mathbf{A}$)	row space
\ker	\ker(\mathbf{A}) → ker($\mathbf{A}$)	kernel
\SD	\SD(X) → SD(X)	standard deviation
\Var	\Var(X) → Var(X)	variation
\Mode	\Mode(X) → Mode(X)	mode
\Median	\Median(X) → Median(X)	median
\gcd	\gcd(X) → gcd(X)	lowercase greatest common divisor
\lcm	\lcm(X) → lcm(X)	lowercase lowest common multiple
\GCD	\GCD(X) → GCD(X)	uppercase greatest common divisor
\LCM	\LCM(X) → LCM(X)	uppercase lowest common multiple
\UnitVector	\UnitVector(\mathbf{r}) → UnitVector($\mathbf{r}$)	unit vector
\principalvalue	\pv{\int f(z) \dd{z}}→ $\mathcal{P}\int f(z)\,dz$	Cauchy principal value
	\PV{\int f(z) \dd{z}}→ P.V. $\int f(z)\,dz$	alternate

5.4 Utilities

\mathcolorbox	\mcbox{color}{content} \mcbox{cyan}{\typical} → ■	\colorbox for math environment, applying to all four levels of math styles
\autocolorbox or \acbox	\cbox{color}{content}	calls \colorbox when in text mode, calls \mathcolorbox when in math mode

`\tentothepowerof`	`\tenpow{n}` → 10^n	work in both math mode and text mode
`\scientificnotation`	`\scinote{3.00}{8}` → 3.00×10^8	work in both math mode and text mode
`\numbercircled`	`\numcir{1}` → ①	patched `\textcircled` for numbers
`\boldsymbol`	`\bsb{\tau}` → $\boldsymbol{\tau}$	shorthand for `\boldsymbol`
`\RNum`	`\RNum{1}` → I	uppercase roman numeral
`\flatfrac`	`\flatfrac{a}{b}` → a/b	flat fraction

5.5 Quick quad text

This set of commands produces text in math-mode padded by `\quad` spacing on either side. This is meant to provide a quick way to insert simple words or phrases in a sequence of equations. Each of the following commands includes a starred version which pads the text only on the right side with `\quad` for use in aligned environments such as `cases`.

General text:		
`\qqtext`	`\qq{}`	general quick quad text with argument
	`\qq{word or phrase}` → _word or phrase_	normal mode; left and right `\quad`
	`\qq*{word or phrase}` → word or phrase_	starred mode; right `\quad` only

Special macros:	
`\qcomma` or `\qc` →,_	right `\quad` only
`\qcc` →_c.c._	complex conjugate; left and right `\quad` unless starred `\qcc*` → c.c._
`\qif` →_if_	left and right `\quad` unless starred `\qif*` → if_

Similar to `\qif`:
`\qthen`, `\qelse`, `\qotherwise`, `\qunless`, `\qgiven`, `\qusing`, `\qassume`, `\qsince`, `\qlet` `\qfor`, `\qall`, `\qeven`, `\qodd`, `\qinteger`, `\qand`, `\qor`, `\qas`, `\qin`

5.6 Derivatives

The default differential symbol d which is used in `\differential` and `\derivative` can be switched to an italic form *d* by including the option `italicdiff` in the preamble:

```
\usepackage[italicdiff]{physics-patch}
```

`\differential`	`\dd` → d `\dd x` → $\mathrm{d}x$ `\dd{x}` → $\,\mathrm{d}x\,$ `\dd[3]{x}` → d^3x `\dd(\cos\theta)` → $\mathrm{d}(\cos\theta)$	no spacing (not recommended) automatic spacing based on neighbors optional power long-form; automatic braces
`\PTderivative`	`\PTdv{x}` → $\frac{\mathrm{d}}{\mathrm{d}x}$ `\PTdv{f}{x}` → $\frac{\mathrm{d}f}{\mathrm{d}x}$ `\PTdv[n]{f}{x}` → $\frac{\mathrm{d}^n f}{\mathrm{d}x^n}$ `\PTdv{x}(\grande)` → $\frac{\mathrm{d}}{\mathrm{d}x}\left(\blacksquare\right)\left(\blacksquare\right)$ `\PTdv*{f}{x}` → $\mathrm{d}f/\mathrm{d}x$	one argument two arguments optional power long-form; automatic braces, spacing inline form using `\flatfrac`
`\PTpartialderivative` or `\PTpderivative`	`\PTdv{f}{x}(\grande)` → $\frac{\mathrm{d}f}{\mathrm{d}x}\left(\blacksquare\right)$ `\PTpdv{x}` → $\frac{\partial}{\partial x}$ `\PTpdv{f}{x}` → $\frac{\partial f}{\partial x}$ `\PTpdv[n]{f}{x}` → $\frac{\partial^n f}{\partial x^n}$ `\PTpdv{x}(\grande)` → $\frac{\partial}{\partial x}\left(\blacksquare\right)$ `\PTpdv{f}{x}{y}` → $\frac{\partial^2 f}{\partial x \partial y}$ `\PTpdv*{f}{x}` → $\partial f/\partial x$ `\PTpdv{f}{x}(\grande)` → $\frac{\partial f}{\partial x}\left(\blacksquare\right)$	note: in original `physics` package, `\dv{f}{x}(\grande)` → $\frac{\mathrm{d}f}{\mathrm{d}x}$ shorthand name two arguments optional power long-form mixed partial inline form using `\flatfrac` note: in original `physics` package, `\pdv{f}{x}(\grande)` → $\frac{\partial f}{\partial x}$
`\variation`	`\var{F[g(x)]}` → $\delta F[g(x)]$ `\var(E-TS)` → $\delta(E-TS)$	functional variation (works like `\dd`) long-form
`\functionalderivative`	`\fdv{g}` → $\frac{\delta}{\delta g}$ `\fdv{F}{g}` → $\frac{\delta F}{\delta g}$ `\fdv{V}(E-TS)` → $\frac{\delta}{\delta V}(E-TS)$ `\fdv*{F}{x}` → $\delta F/\delta x$	functional derivative (works like `\PTdv`) long-form inline form using `\flatfrac`

5.7 Dirac bra-ket notation

The following collection of macros for Dirac notation contains two fundamental commands, `\bra` and `\ket`, along with a set of more specialized macros which are essentially combinations of the fundamental pair. The fundamental commands are designed to contract with one another algebraically when appropriate and are thus suggested for general use. For instance, the following code renders correctly[1]

`\bra{\phi}\ket{\psi}` → $\langle\phi|\psi\rangle$ as opposed to $\langle\phi|\psi\rangle$

whereas a similar construction with higher-level macros will not contract in a robust manner

`\bra{\phi}\dyad{\psi}{\xi}` → $\langle\phi|\,|\psi\rangle\langle\xi|\,.$

On the other hand, the correct output can be generated by sticking to the fundamental commands,

`\bra{\phi}\ket{\psi}\bra{\xi}` → $\langle\phi|\psi\rangle\langle\xi|$

allowing the user to type out complicated quantum mechanical expressions without worrying about bra-ket contractions. That being said, the high-level macros do have a place in convenience and readability, as long as the user is aware of rendering issues that may arise due to an absence of automatic contractions.

`\ket`	`\ket{\tall}` → $\left\vert\blacksquare\right\rangle$ `\ket*{\tall}` → $\vert\blacksquare\rangle$	automatic sizing no resize

`\bra`	`\bra{\tall}` → $\left\langle\blacksquare\right\vert$
	`\bra*{\tall}` → $\langle\blacksquare\vert$
	`\bra{\phi}\ket{\psi}`
	→ $\langle\phi\vert\psi\rangle$
	`\bra{\phi}\ket{\tall}`
	→ $\left\langle\phi\middle\vert\blacksquare\right\rangle$
	`\bra{\phi}\ket*{\tall}`
	→ $\langle\phi\vert\blacksquare\rangle$
	`\bra*{\phi}\ket{\tall}`
	→ $\langle\phi\vert\blacksquare\rangle$
	`\bra*{\phi}\ket*{\tall}`
	→ $\langle\phi\vert\blacksquare\rangle$
		automatic sizing
		no resize
		automatic contraction
		
		contraction inherits automatic sizing
		a star on either term in the contraction prohibits resizing

`\innerproduct`	`\braket{a}{b}` → $\langle a\vert b\rangle$
	`\braket{a}` → $\langle a\vert a\rangle$
	`\braket{a}{\tall}`
	→ $\left\langle a\middle\vert\blacksquare\right\rangle$
	`\braket*{a}{\tall}`
	→ $\langle a\vert\blacksquare\rangle$
	`\ip{a}{b}` → $\langle a\vert b\rangle$
		two-argument braket
		one-argument (norm)
		automatic sizing
		
		no resize
		
		shorthand name

[1] Note the lack of a space between the bra and ket commands. This is necessary in order for the bra to find the corresponding ket and form a contraction.

\outerproduct	\dyad{a}{b} → $\lvert a\rangle\langle b\rvert$	two-argument dyad
	\dyad{a} → $\lvert a\rangle\langle a\rvert$	one-argument (projector)
	\dyad{a}{\tall} → $\left\lvert a\right\rangle\left\langle \blacksquare\right\rvert$	automatic sizing
	\dyad*{a}{\tall} → $\lvert a\rangle\langle \blacksquare\rvert$	no resize
	\ketbra{a}{b} → $\lvert a\rangle\langle b\rvert$	alternative name
	\op{a}{b} → $\lvert a\rangle\langle b\rvert$	shorthand name
\expectationvalue	\expval{A} → $\langle A\rangle$	implicit form
	\expval{A}{\Psi} → $\langle\Psi\lvert A\rvert\Psi\rangle$	explicit form
	\ev{A}{\Psi} → $\langle\Psi\lvert A\rvert\Psi\rangle$	shorthand name
	\ev{\grande}{\Psi} → $\langle\Psi\rvert\blacksquare\lvert\Psi\rangle$	default sizing ignores middle argument
	\ev*{\grande}{\tall} → $\langle\blacksquare\rvert\blacksquare\lvert\blacksquare\rangle$	single star does no resizing whatsoever
	\ev**{\grande}{\Psi} → $\left\langle\Psi\right\rvert\blacksquare\left\lvert\Psi\right\rangle$	double star resizes based on all parts
\matrixelement	\matrixel{n}{A}{m} → $\langle n\lvert A\rvert m\rangle$	requires all three arguments
	\mel{n}{A}{m} → $\langle n\lvert A\rvert m\rangle$	shorthand name
	\mel{n}{\grande}{m} → $\langle n\rvert\blacksquare\lvert m\rangle$	default sizing ignores middle argument
	\mel*{n}{\grande}{\tall} → $\langle n\rvert\blacksquare\lvert\blacksquare\rangle$	single star does no resizing whatsoever
	\mel**{n}{\grande}{m} → $\left\langle n\right\rvert\blacksquare\left\lvert m\right\rangle$	double star resizes based on all parts

5.8 Matrix macros

Note: \mqty and \smqty in physics uses \mathord, while \PTmqty and \PTsmqty in physics-patch don't.

The following matrix macros produce unformatted rows and columns of matrix elements for use as separate matrices as well as blocks within larger matrices. For example, the command \identitymatrix{2} which has also has the shortcut \imat{2} produces the elements of a 2×2 identity matrix $\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}$ without braces or grouping. This allows the command to also be used within another matrix, as in:

```
\begin{pmatrix}
\imat{2} \\ a & b
\end{pmatrix}
```

⇒ $\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ a & b \end{pmatrix}$

To specify elements on the right of left sides of our \imat{2} sub-matrix we use the grouping command \PTmatrixquantity or \PTmqty to effectively convert \imat{2} into a single matrix element of a larger matrix:

```
\begin{pmatrix}
\PTmqty{\imat{2}} & \PTmqty{a\\b} \\ \PTmqty{c & d} &
    e
\end{pmatrix}
```
$\Rightarrow$ $\begin{pmatrix} 1 & 0 & a \\ 0 & 1 & b \\ c & d & e \end{pmatrix}$

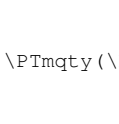
The extra `\PTmqty` groups were required in this case in order to get the a and b elements to behave as a single element, since `\PTmqty{\imat{2}}` also acts like a single matrix element (the same can be said of the grouped c and d elements). Finally, the outermost `pmatrix` environment could have also been replaced with the `physics-patch` macro `\PTmqty()`, allowing the above example to be written on one line:

```
\PTmqty(\PTmqty{\imat{2}} & \PTmqty{a\\b} \\ \PTmqty{
    c & d} & e)
```
$\Rightarrow$ $\begin{pmatrix} 1 & 0 & a \\ 0 & 1 & b \\ c & d & e \end{pmatrix}$

`\PTmatrixquantity`	`\PTmqty{a & b \\ c & d}` → $\begin{matrix} a & b \\ c & d \end{matrix}$	groups a set of matrix elements into a single object
	`\PTmqty(a & b \\ c & d)` → $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$	parentheses
	`\PTmqty*(a & b \\ c & d)` → $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$	alternate parentheses
	`\PTmqty[a & b \\ c & d]` → $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$	square brackets
	`\PTmqty | a & b \\ c & d |` → $\begin{vmatrix} a & b \\ c & d \end{vmatrix}$	vertical bars

	`\omqty{}` ↔ `\PTmqty{}`
	`\pmqty{}` ↔ `\PTmqty()`
	`\PTpmqty{}` ↔ `\PTmqty()`
	`\PTpmqty*{}` ↔ `\PTmqty*{}`
	`\Pmqty{}` ↔ `\PTmqty*()`
	`\bmqty{}` ↔ `\PTmqty[]`
	`\vmqty{}` ↔ `\PTmqty||`
		alternative syntax; robust and more LaTeX-friendly

`\PTsmallmatrixquantity`	`\PTsmqty{a & b \\ c & d}` → $\left(\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right)$	the `smallmatrix` form of `\PTmqty`
	`\PTsmqty()` or `\spmqty{}` or `\PTspmqty{}`	small version of `\PTmqty()`
	`\PTsmqty*()` or `\sPmqty{}` or `\PTspmqty*{}`	small version of `\PTmqty*()`
	`\PTsmqty[]` or `\sbmqty{}`	small version of `\PTmqty[]`
	`\PTsmqty||` or `\svmqty{}`	small version of `\PTmqty||`
`\matrixdeterminant`	`\mdet{a & b \\ c & d}` → $\begin{vmatrix} a & b \\ c & d \end{vmatrix}$	matrix determinant
	`\smdet{a & b \\ c & d}` → $\left\lvert\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}\right\rvert$	small matrix determinant
`\identitymatrix`	`\imat{n}`	elements of $n \times n$ identity matrix
	`\PTmqty(\imat{3})` → $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	formatted with `\PTmqty` or `\PTsmqty`
`\PTxmatrix`	`\PTxmat{x}{n}{m}`	elements of $n \times m$ matrix filled with x, if not provided, 1 is used
	`\PTmqty(\PTxmat{x}{3}{3})` → $\begin{pmatrix} x & x & x \\ x & x & x \\ x & x & x \end{pmatrix}$	
	`\PTmqty(\PTxmat{x}{}{3})` → $\begin{pmatrix} x & x & x \end{pmatrix}$	

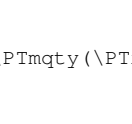
`\PTmqty(\PTxmat{x}{3}{})` → 

`\PTxmat*{x}{n}{m}`

`\PTmqty(\PTxmat*{x}{3}{3})` →

star for element indices, skip row/column indices $n = 1/m = 1$

`\PTmqty(\PTxmat*{x}{1}{3})` →

`\PTmqty(\PTxmat*{x}{3}{1})` → 

`\PTxmat{x}{n}{m}[p]`

only show p rows (including `\vdots` row) with skipped rows indicated by `\vdots`. If n isn't provided, p is used

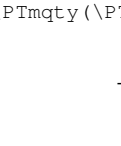
`\PTmqty(\PTxmat{x}{5}{3}[3])` →

`\PTmqty(\PTxmat{x}{3}{3}[3])` →

`\PTmqty(\PTxmat{x}{}{3}[3])` →

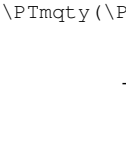
`\PTxmat{x}{n}{m}[p][q]`

only show p rows (including `\vdots` row) and q columns (including `\ldots` column) with skipped rows indicated by `\vdots`, skipped columns indicated by `\ldots`, intersection of `\vdots` row and `\ldots` column being `\ddots`. If n/m isn't provided, p/q is used. No indices will be added for ellipses even if star is given

`\PTmqty(\PTxmat{x}{5}{5}[3][3])` → 

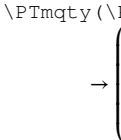
`\PTmqty(\PTxmat{x}{5}{3}[3][3])` →

`\PTmqty(\PTxmat{x}{3}{3}[3][3])` →

`\PTmqty(\PTxmat{x}{5}{}[3][3])` → 

`\PTmqty(\PTxmat{x}{}{}[3][3])` →

`\PTxmat*{x}{n}{m}{g}`

`\PTmqty(\PTxmat*{x}{3}{3}{A})` → 

customize last row's element indices to g

	`\PTmqty(\PTxmat*{x}{5}{5}[3][3]{A})` $\rightarrow \begin{pmatrix} x_{11} & \cdots & x_{15} \\ \vdots & \ddots & \vdots \\ x_{A1} & \cdots & x_{A5} \end{pmatrix}$ `\PTxmat*{x}{n}{m}{g}{h}` `\PTmqty(\PTxmat*{x}{3}{3}{A}{B})` $\rightarrow \begin{pmatrix} x_{11} & x_{12} & x_{1B} \\ x_{21} & x_{22} & x_{2B} \\ x_{A1} & x_{A2} & x_{AB} \end{pmatrix}$ `\PTmqty(\PTxmat*{x}{5}{5}[3][3]{A}{B})` $\rightarrow \begin{pmatrix} x_{11} & \cdots & x_{1B} \\ \vdots & \ddots & \vdots \\ x_{A1} & \cdots & x_{AB} \end{pmatrix}$	customize last row's element indices to g and last column's element indices to h
	`\PTxmat[0 or 1 or 2]{x}{n}{m}[p][q]` `\PTmqty(\PTxmat[0]{x}{5}{5}[3][3])` $\rightarrow \begin{pmatrix} x & x & \cdots \\ x & x & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}$ `\PTmqty(\PTxmat[1]{x}{5}{5}[3][3])` $\rightarrow \begin{pmatrix} x & \cdots & x \\ x & \cdots & x \\ \vdots & \ddots & \vdots \end{pmatrix}$ `\PTmqty(\PTxmat[2]{x}{5}{5}[3][3])` $\rightarrow \begin{pmatrix} x & x & \cdots \\ \vdots & \vdots & \ddots \\ x & x & \cdots \end{pmatrix}$	Change the `\vdots` row/`\ldots` column from the second last one to last one, 0 for both, 1 for row only, 2 for column only. Only work when corresponding p/q is provided and do not change the behavior of element indices
`\zeromatrix`	`\zmat{n}{m}` `\PTmqty(\zmat{2}{2})` $\rightarrow \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$ `\PTmqty(\zmat{2})` $\rightarrow \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$	$n \times m$ matrix filled with zeros, equivalent to `\xmat{0}{n}{m}`. If m isn't provided, n is used
`\paulimatrix`	`\pmat{n}` `\PTmqty(\pmat{0})` $\rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ `\PTmqty(\pmat{1})` $\rightarrow \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ `\PTmqty(\pmat{2})` $\rightarrow \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$ `\PTmqty(\pmat{3})` $\rightarrow \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	n^{th} Pauli matrix $n \in \{0, 1, 2, 3 \text{ or } x, y, z\}$
`\diagonalmatrix`	`\dmat{a,b,c,...}` `\PTmqty(\dmat{1,2,3})` $\rightarrow \begin{pmatrix} 1 & & \\ & 2 & \\ & & 3 \end{pmatrix}$ `\PTmqty(\dmat[0]{1,2})` $\rightarrow \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$ `\PTmqty(\dmat{1,2&3\\4&5})` $\rightarrow \begin{pmatrix} 1 & & \\ & 2 & 3 \\ & 4 & 5 \end{pmatrix}$	specify up to eight diagonal or block diagonal elements optional argument to fill spaces enter matrix elements for each block as a single diagonal element
`\antidiagonalmatrix`	`\admat{a,b,c,...}`	same as syntax as `\dmat`

	\PTmqty(\admat{1,2,3}) → $\begin{pmatrix} & & 1 \\ & 2 & \\ 3 & & \end{pmatrix}$	

5.9 Symbols

\lparen → (	
\rparen →)	
\ordersymbol → $\mathcal{O}$	
\typical →	
\tall →	
\grande →	
\venti →	
\parallelsum →/	
\calE→\mathcal{E}	
\bbR→\mathbb{R}	
\bbC→\mathbb{C}	
\bbQ→\mathbb{Q}	
\bbN→\mathbb{N}	
\bbZ→\mathbb{Z}	
\bell→\boldsymbol{\ell}	
\Bell→\char"1F514	the \bell command in wasysym, which is a bell symbol
\Vtextvisiblespace[width] → ␣	a visible space character, where the optional argument, defaulting to .3em, sets the width of the horizontal rule

5.10 Arrows and lines

Note that only \equiv and \eqv work in both math mode and text mode; others are for math mode only.

\Leftrightarrow or \Lra	⇔
\leftrightarrow or \lra	↔
\Rightarrow or \Ra	⇒
\rightarrow or \ra	→
\Leftarrow or \La	⇐
\leftarrow or \la	←
\Uparrow or \Upa	⇑
\uparrow or \upa	↑
\Downarrow or \Dna	⇓
\downarrow or \dna	↓

`\rightleftharpoons` or `\rlh`	$\rightleftharpoons$
`\leftrightharpoons` or `\lrh`	$\leftrightharpoons$
`\rightharpoonup` or `\rhu`	$\rightharpoonup$
`\leftharpoonup` or `\lhu`	$\leftharpoonup$
`\rightharpoondown` or `\rhd`	$\rightharpoondown$
`\leftharpoondown` or `\lhd`	$\leftharpoondown$
`\upharpoonright` or `\uhr`	$\upharpoonright$
`\upharpoonleft` or `\uhl`	$\upharpoonleft$
`\downharpoonright` or `\dhr`	$\downharpoonright$
`\downharpoonleft` or `\dhl`	$\downharpoonleft$
`\hookrightarrow` or `\hkra`	$\hookrightarrow$
`\hookleftarrow` or `\hkla`	$\hookleftarrow$
`\mapsto` or `\mpto`	$\mapsto$
`\mapsfrom` or `\mpfr`	$\mapsfrom$
`\equiv` or `\eqv`	$\equiv$
`\stackrel{\mathrm{def}}{=}` or `\defeq`	$\stackrel{\mathrm{def}}{=}$

5.11 Shorthands for Greek alphabet

If the corresponding options are used, the following shorthands will be defined for every uppercase and lowercase Greek letter. Note that these don't ensure those commands are defined. Take Alpha for example.

Command	Option	Note
`\tgAlpha` → `\text{\textAlpha}}`	shorttextgreek	accept an optional argument argument in `{}` that is simply skipped
`\vAlpha` → `\varAlpha`	shortvargreek	
`\uAlpha` → `\upAlpha`	shortupgreek	
`\uvAlpha` → `\upvarAlpha`	shortupvargreek	
`\bAlpha` → `\boldsymbol{\Alpha}`	shortboldgreek	

5.12 Shorthands for mathrm alphabet and chemical element symbols

If option `shortmathrm` is used, the following shorthands will be defined for every uppercase and lowercase English letter and every chemical element symbols. Take A for example.

`\rmA_a^b` → A_a^b	work in both math mode and text mode

which are implemented with:

```
\DeclareDocumentCommand{\rmA}{e{_^}}{\ensuremath{\mathrm{A}\IfValueTF{#1}{_{#1}}{}\IfValueTF{#2}{^{#2}}{}}}
```

5.13 Shorthands for textnormal alphabet

If option `shorttext` is used, the following shorthands will be defined for every uppercase and lowercase English letter. Take A for example.

`\txA` → `\textnormal{A}`