# The physics package

Sergio C. de la Barrera physics.tex@gmail.com

November 1, 2012

## Contents

1	Bef	ore you start	1
	1.1	The purpose of this package	1
	1.2	Other required packages	1
	1.3	Using physics in your LATEX document	2
2		of commands	2
	2.1	Automatic bracing	2
		Vector notation	
	2.3	Operators	3
	2.4	Quick quad text	3
	2.5	Derivatives	4
	2.6	Dirac bra-ket notation	5

## 1 Before you start

#### 1.1 The purpose of this package

The goal of this package is to make typesetting equations for physics simpler, faster, and more humanreadable. To that end, the commands included in this package have names that make the purpose of each command immediately obvious and remove any ambiguity while reading and editing physics code. From a practical standpoint, it is handy to have a well-defined set of shortcuts for accessing the long-form of each of these commands. The commands listed below are therefore defined in terms of their long-form names and then shown explicitly in terms of the default shorthand command sequences. These shorthand commands are meant make it easy to remember both the shorthand names and what each one represents.

#### 1.2 Other required packages

The physics package requires xparse and amsmath to work properly in your LATEX document. The amsmath package comes standard with most LATEX distributions and is loaded by physics for your convenience. You may also already have xparse installed on your system as it is a popular package for defining LATEX macros, however, if you are unsure you can either install it again using your local package manager (comes with most distributions) or by visiting the CTAN online package database, or you could even just try to use physics without worrying about it. Many modern LATEX compilers will locate and offer to download missing packages for you.

## 1.3 Using physics in your LATEX document

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

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

# 2 List of commands

## 2.1 Automatic bracing

\quantity	$\label{eq:constraint} \begin{array}{l} \langle qty(a+b) \rightarrow (a+b) \\ \langle qty[a+b] \rightarrow [a+b] \\ \langle qty[a+b] \rightarrow [a+b] \\ \langle qty\{a+b\} \rightarrow \{a+b\} \\ \langle qty \ big\{\} \rightarrow \{\} \\ \langle qty \ big\{\} \rightarrow \{\} \\ \langle qty \ bigg\{\} \ bigg$	automatic () braces automatic [] braces automatic    braces automatic { } braces manual sizing (works with any of the above bracket types)
\absolutevalue	$abs{a} \rightarrow  a $	automatic sizing; equivalent to $ qty a $
	abs   Big a > a	inherits manual sizing syntax from \qty
\norm	$\texttt{\norm{a}} \to \ a\ $	automatic sizing
	$\operatorname{Norm}Big{a} \rightarrow a$	manual sizing
\evaluated	$\texttt{Veval{x}_0^{infty}} \xrightarrow{\parallel} x _0^\infty$	automatic right vertical bar for evalua- tion limits
	$(x)_y \rightarrow (x)_u$	parenthetical form for holding values
	0	constant
\order	$\operatorname{rfx^2}  o \mathcal{O}\!\left(x^2 ight)$	order symbol; automatic sizing and space handling
	$\operatorname{Sig}{x^2}  o \mathcal{O}(x^2)$	manual sizing
\commutator	$\operatorname{Comm}{A}{B} \rightarrow [A, B]$	automatic sizing
\anticommutator	$\lambda = \{A\} \{B\} \rightarrow \{A, B\}$	same as \poissonbracket
	$\texttt{A} \in \{A, B\} \rightarrow \{A, B\}$	alternate name
poissonbracket	$\mathbf{B} \to \{A, B\} \to \{A, B\}$	same as $\anticommutator$

## 2.2 Vector notation

\vectorbold	$\texttt{vb}\{\mathtt{a}\} \to \mathtt{a}$	upright/no Greek
	\vb*{a}, \vb*{\theta} $ ightarrow oldsymbol{a}, oldsymbol{ heta}$	italic/Greek

\vectorarrow	$va{a} \rightarrow \vec{a}$	upright/no Greek
	$\mathbf{x}_{a}, \mathbf{v}_{a} \in \mathbf{d}, \mathbf{d}$	italic/Greek
\vectorunit	$vu{a}  ightarrow {\hat{a}}$	upright/no Greek
	$\forall u \in a, \forall u \in \{ theta \}  ightarrow \hat{m{a}}, \hat{m{ heta}}$	italic/Greek
\dotproduct	$\forall dot \rightarrow \cdot as in a \cdot b$	note that $dp$ is a protected TEX primitive
$\crossproduct$	$cross \rightarrow x as in a \times b$	
	$\c p \rightarrow \times as in a \times b$	alternate name
\vnabla	$\forall nabla  ightarrow oldsymbol{ abla} \; versus   abla$	low-level macro for bold version
\gradient	$\grad  ightarrow {m  abla}$	
	$\texttt{Psi} \to \boldsymbol{\nabla} \Psi$	default mode
	$\texttt{Vgrad}(\texttt{Psi+Phi}) \rightarrow \boldsymbol{\nabla}(\Psi + \Phi)$	long-form (like \qty but also handles spacing)
	$\left[ Psi+Phi \right]  ightarrow oldsymbol{ abla} \Psi + \Phi ]$	
\divergence	\div $ ightarrow {f  abla} \cdot$	note $\texttt{amsmath}$ symbol $\div$ renamed
		\divisionsymbol
	$\det\{\nabla_a\} \to {oldsymbol  abla} \cdot {f a}$	default mode
	$\det(\nabla_{a}+\nabla_{b}) \rightarrow {oldsymbol  abla} \cdot ({f a}+{f b})$	long-form
	$\det[\nabla b{a}+\nabla b{b}] \rightarrow \nabla \cdot [a+b]$	
\curl	$\operatorname{Curl}  ightarrow {oldsymbol  abla}  imes$	
	$\texttt{vurl} \to \boldsymbol{\nabla} \times \mathbf{a}$	default mode
	$\texttt{(vb{a}+vb{b})} \rightarrow \boldsymbol{\nabla} \times (\mathbf{a} + \mathbf{b})$	long-form
	$\texttt{vurl[vb{a}+vb{b}]} \rightarrow \boldsymbol{\nabla} \times [\mathbf{a} + \mathbf{b}]$	
\laplacian	$lapha$ laplacian $ o  abla^2$	
	<code>\laplacian{\Psi} <math> ightarrow  abla^2 \Psi</math></code>	default mode
	<code>\laplacian(\Psi+\Phi)</code> $ ightarrow  abla^2(\Psi + \Phi)$	long-form
	$\lambda = \nabla^2 [\Psi + \Phi]$	

### 2.3 Operators

The standard set of trig functions is redefined in physics to provide automatic braces that behave like qty(). In addition, an optional power argument is provided. The old definition of each trig function is stored in a long form version as in  $sin \rightarrow sine$ ,  $cos \rightarrow cosine$ , and so on.

Trig redefinitions:			
\sin	$\operatorname{sin}(\mathbf{x}) \to \sin(x)$	automatic braces; old \sin renamed \sine	
	$sin[2](x) \rightarrow sin^2(x)$	optional power	
\cos	$\cos(x) \rightarrow \cos(x)$	works with all trig functions	
\tan	$\tan x \to \tan x$	can still use without an argument	
÷		:	
New operators:			
\tr	$\texttt{tr} \rho \rightarrow \operatorname{tr} \rho$	trace	
\rank	$\texttt{rank } M \to \mathrm{rank} M$	matrix rank	
\erf	$\operatorname{verf}(\mathbf{x}) \to \operatorname{erf}(x)$	Gauss error function	

### 2.4 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	$qq{} \rightarrow word or phrase} \rightarrow word or phrase / qq{word or phrase} \rightarrow word or phrase / qq*{word or phrase}$	general quick quad text with argument normal mode; left and right starred mode; right  only
Special macros	:	

$qcomma \text{ or }qc \rightarrow, \_$	right  only
$\verb+qcc \rightarrow \_c.c.\_$	complex conjugate; left and right  unless starred \qcc* $\rightarrow$ c.c.
$\fi \rightarrow _if$	left and right  unless starred \qif* $\rightarrow$ if

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

## 2.5 Derivatives

\differential	$dd \rightarrow d$	
	\dd $\mathbf{x} \to \mathrm{d}x$	no spacing; use in braces, fractions
	$dd{x} \rightarrow dx$	proper spacing for typical equations
	\dd[3]{x} $ ightarrow \mathrm{d}^3x$	optional power
	$\operatorname{dd}(\operatorname{cos}) \to \operatorname{d}(\cos\theta)$	long-form; automatic braces
\derivative		one argument
	$\det{f}{x} \rightarrow \frac{\mathrm{d}f}{\mathrm{d}x}$	two arguments
	$\operatorname{dv}[n]{f}{x} \rightarrow \frac{\mathrm{d}^n J}{\mathrm{d} x_1^n}$	optional power
	$\operatorname{dv}{x}(x^2+x) \rightarrow \frac{\mathrm{d}}{\mathrm{d}x}(x^2+x)$	long-form; automatic braces, spacing
\partialderivative	$ extsf{vderivative{x}}  o rac{\partial}{\partial x}$	alternate name
	$\t x \rightarrow \frac{\partial}{\partial x} $	shorthand name
	$pdv{f}{x} \rightarrow \frac{\partial f}{\partial x} = \frac{\partial f}{\partial x}$	two arguments
	$\operatorname{pdv}[n]{f}{x} \rightarrow \frac{\partial f}{\partial x^n}$	optional power
	$\operatorname{ydv}{x}(x+y) \to \frac{\partial}{\partial x}(x+y)$	long-form
	$pdv{f}{x}{y} \rightarrow \frac{\partial f}{\partial x \partial y}$	mixed partial
\variation	$\operatorname{Var}{F[g(x)]} \rightarrow \delta F[g(x)]$	functional variation (works like \dd)
	$\operatorname{Var}(E-TS) \rightarrow \delta(E-TS)$	long-form
\functionalderivative	$ \begin{array}{l} (\operatorname{var}(\operatorname{E-TS}) \to \delta(E - TS) \\ \operatorname{var}(\operatorname{E-TS}) \to \delta(E - TS) \\ \operatorname{var}(\operatorname{g}) \to \frac{\delta}{\delta g} \\ \operatorname{var}(\operatorname{F}) \{g\} \to \frac{\delta F}{\delta g} \\ \operatorname{var}(\operatorname{F}) \{g\} \to \frac{\delta F}{\delta g} \\ \operatorname{var}(\operatorname{F}) \{g\} \to \frac{\delta}{\delta V}(E - TS) \end{array} $	functional derivative (works like $\ \vec{dv})$
	$fdv{F}{g}  o {\delta F'\over \delta g}$	
	$fdv{V}(E-TS) \rightarrow \frac{\delta}{\delta V}(E-TS)$	long-form

#### 2.6 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 specialized macros are both useful and descriptive from the perspective of generating physics code, however, 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<sup>1</sup>

$$\mathbf{\Phi} \in \{\phi | \psi\} \text{ as opposed to } \langle \phi | \psi \rangle$$

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

$$\mathbf{\Phi} = \langle \phi | | \psi \rangle \langle \xi |$$

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

$$\bar{\psi} \langle \xi | \psi \rangle \langle \xi |$$

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.

$\label{eq:product} \begin{array}{llllllllllllllllllllllllllllllllllll$	\ket	\ket{\psi} $ ightarrow  \psi angle$	automatic sizing
$eq:linear_line$		$\texttt{ket*{\psi}} \to \langle \psi  $	complex conjugate (looks like <b>\bra</b> but
$eq:linear_line$			,
$\label{eq:constraint} \begin{array}{llllllllllllllllllllllllllllllllllll$	\bra		0
$\label{eq:phi}ket{\psi} \rightarrow \langle \phi   \psi \rangle  \mbox{automatic contraction} \\ \mbox{braket{a}{b}} \rightarrow \langle a   b \rangle  \mbox{two-argument contraction; automatic sizing} \\ \mbox{braket{a}} \rightarrow \langle a   a \rangle  \mbox{single-argument; produces norm} \\ \mbox{braket*{a}{b}} \rightarrow \langle b   a \rangle  \mbox{complex conjugate; swaps arguments} \\ \mbox{ip{a}{b}} \rightarrow \langle a   b \rangle  \mbox{shorthand name} \\ \mbox{voterproduct}  \mbox{dyad{a}{b}} \rightarrow  a \rangle \langle b    \mbox{two-argument dyad; automatic sizing} \\ \mbox{dyad{a}} \rightarrow  a \rangle \langle a    \mbox{single-argument; produces projector} \\ \mbox{dyad{a}} \rightarrow  a \rangle \langle a    \mbox{single-argument; produces projector} \\ \mbox{dyad{a}} \rightarrow  a \rangle \langle a    \mbox{single-argument; produces projector} \\ \mbox{dyad{a}} \ \mbox{dyad{a}} \rightarrow  a \rangle \langle b    \mbox{alternative name} \\ \mbox{lexpectationvalue}  \box{lexpval{b}} \rightarrow  a \rangle \langle b    \mbox{shorthand name} \\ \mbox{lexpval{A}} \ \mbox{dyal{A}} \ \mbox{dyad{}} \ \mbox{shorthand name} \\ \mbox{lexpval{A}} \ \mbox{dyad{}} \ \mbox{dyad{}} \ \mbox{shorthand name} \\ \mbox{lexpval{A}} \ \mbox{dyad{}} \ \mbox{dyad{}} \ \mbox{dyad{}} \ \mbox{shorthand name} \\ \mbox{lexpval{A}} \ \mbox{dyad{}} \ \mbox{dyad{}} \ \mbox{shorthand name} \ \mbox{lexpval{A}} \ \mbox{dyad{}} \ \mbox{dyad{}} \ \mbox{shorthand name} \ \mbox{lexpval{A}} \ \mbox{lexpval{A}} \ \mbox{dyad{}} \ \mbox{shorthand name} \ \mbox{lexpval{A}} \ \mbox{lexpval{A}} \ \mbox{lexpval{A}} \ \mbox{dyad{}} \ \mbox{shorthand name} \ \mbox{lexpval{A}} \ \mbox{lexpval{A}} \ \mbox{lexpval{A}} \ \mbox{lexpval{A}} \ \mbox{lexpval{A}} \ \mbox{shorthand name} \ \mbox{lexpval{A}} \ \mbox{lexpval{A}} \ \mbox{shorthand name} \ \mbox{lexpval{A}} \$		\bra*{\psi} $ ightarrow  \psi angle$	- • • • •
$\label{eq:linerproduct} $$ \braket{a}{b} \rightarrow \langle a b\rangle$ two-argument contraction; automatic sizing $$ \braket{a} \rightarrow \langle a a\rangle$ single-argument; produces norm $$ \braket*{a}{b} \rightarrow \langle b a\rangle$ complex conjugate; swaps arguments $$ \ip{a}{b} \rightarrow \langle a b\rangle$ shorthand name $$ \braket{a}{b} \rightarrow  a\rangle\langle b $ two-argument dyad; automatic sizing $$ \dyad{a}{b} \rightarrow  a\rangle\langle b $ two-argument dyad; automatic sizing $$ \dyad{a}{b} \rightarrow  a\rangle\langle a $ two-argument; produces projector $$ \dyad{a}{b} \rightarrow  a\rangle\langle a $ two-argument; produces projector $$ \dyad{a}{b} \rightarrow  a\rangle\langle a $ two-argument; produces projector $$ \dyad{a}{b} \rightarrow  a\rangle\langle b $ alternative name $$ \braket{a}{b} \rightarrow  a\rangle\langle b $ alternative name $$ \braket{a}{b} \rightarrow  a\rangle\langle b $ alternative name $$ \braket{a}{b} \rightarrow  a\rangle\langle b $ shorthand name $$ \braket{a}{b} \rightarrow  a\rangle\langle b $ \braket{a}{b} \rightarrow  a\rangle\langle b $ shorthand name $$ \braket{a}{b} \rightarrow  a\rangle\langle b $			,
$\label{eq:sizing} \begin{tabular}{lllllllllllllllllllllllllllllllllll$			automatic contraction
$\label{eq:stability} $$ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \$	$\innerproduct$	\braket{a}{b} $ ightarrow \langle a b angle$	two-argument contraction; automatic
$eq:linear_line$			sizing
$\label{eq:linear} $$ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \$		\braket{a} $ ightarrow \langle a   a  angle$	single-argument; produces norm
$eq:linear_line$			complex conjugate; swaps arguments
$\label{eq:linear} \begin{array}{llllllllllllllllllllllllllllllllllll$			shorthand name
$eq:linear_line$	$\outerproduct$		two-argument dyad; automatic sizing
$\label{eq:linear} $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$$		\dyad{a} $ ightarrow  a angle\!\langle a $	single-argument; produces projector
$\label{eq:logith} $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$ $$$		\dyad*{a}{b} $ ightarrow  b angle\!\langle a $	complex conjugate; swaps arguments
$\label{eq:label} \begin{array}{lllllllllllllllllllllllllllllllllll$		\ketbra{a}{b} $ ightarrow  a angle\!\langle b $	alternative name
$\begin{split} & \texttt{expval}\{\texttt{A}\}\{\texttt{Psi}\} \to \langle \Psi   A   \Psi \rangle & \text{explicit form} \\ & \texttt{ev}\{\texttt{A}\}\{\texttt{Psi}\} \to \langle \Psi   A   \Psi \rangle & \text{shorthand name} \end{split}$		$egin{aligned} \black b \black \bla$	shorthand name
$\exp{A}{\Psi i} \rightarrow \langle \Psi   A   \Psi $ shorthand name	$\ensuremath{expectationvalue}$	$\texttt{A} \rightarrow \langle A \rangle$	implicit form
		$\exp\{A} \in \langle \Psi   A   \Psi \rangle$	explicit form
$\verb+matrixelement \\ \verb+matrixel{n}{A}{m} \rightarrow \langle n A m \rangle  \text{requires all three arguments}$		$\texttt{A}{\Phi} \rightarrow \langle \Psi   A   \Psi \rangle$	shorthand name
	\matrixelement	$\texttt{Matrixel{n}} \rightarrow \langle n   A   m \rangle$	requires all three arguments
$\mathbf{A}_{n} \rightarrow \langle n   A   m \qquad \text{ shorthand name}$		$\texttt{Mel{n}{A}} \rightarrow \langle n A m\rangle$	shorthand name

 $<sup>^{1}</sup>$ Note the lack of a space between the bra and ket commands. This is necessary is order for the bra to find the corresponding ket and form a contraction.