# The revquantum package

Christopher Granade
cgranade@cgranade.com

v0.11 from 2017/03/15

## 1 Introduction

The revquantum package provides a number of useful hacks to solve common annoyances with the revtex4-1 package, and to define notation in common use within quantum information. In doing so, revquantum imports and configures a number of commonly-available and used packages, and where reasonable, provides fallbacks.

The revquantum package also warns when users try to load packages which are known to be incompatible with revtex4-1. In particular, loading the following packages will cause warnings:

- subcaption

Yes, this is a short list. It will get longer.

## 2 Usage

**NB:** revquantum must be loaded first unless `nobibtexhacks` is passed.

### 2.1 Package Options

The revquantum package provides several options to configure its behavior. These can be used in the traditional way, as optional arguments to `\usepackage`. For instance, this document was prepared using `\usepackage[pretty]{revquantum}`.

`[final]`

Removes support for TODO commands (see below), causing them to escalate from warnings to errors.

`[pretty]`

Uses the mathpazo package to typeset the document more nicely than the default for revtex4-1 drafts.

`[uselistings]`

Includes the listings package and configures it for literate and math-escape notation in Python, Mathematica and MATLAB.

`[nobibtexhacks]`

Prevents revquantum from patching the BibTeXsupport provided by revtex4-1 to include titles.

`[strict]`

Causes package incompatibility warnings to become errors.

## 2.2 New Commands

### 2.2.1 TODO Annotations

`\todo` `{`⟨*annotation*⟩`}`

Marks an incompleted task in a different color in the document, and raises a warning in the LaTeX log.

`\TODO` TODO

`\todolist` `{`⟨*contents*⟩`}`

Typesets *contents* as a TODO annotation, wrapped in an `{enumerate}` environment.

### 2.2.2 Mathematical Notation

This package also provides commands for notation common in quantum information.

`\ii`
`\dd`

These commands typeset the imaginary unit i and differential element d, respectively, in math roman.

`\defeq`

This command uses `\mathrel` to properly format the colon-equals operator as a relation operator.

`\expect` Typesets the expectation operator $\mathbb{E}$.

`\id`

The current implementation of id is to call openone, provided by revtex4-1, and thus not available when used from documentation.

`\llbracket`
`\rrbracket`

Typesets the double-square brackets commonly used to denote stabilizer code properties, as in $[\![n, k, d]\!]$. These commands are provided by `{stmaryrd}`, and are given a reasonable fallback if that package is not available.

### 2.2.3 Affiliation Database

The revquantum package provides commands for quickly typesetting affiliations, with an eye towards reducing copy-paste errors when authors have a nontrivial set of shared affiliations.

`\newaffil` `{`⟨*shorthand*⟩`}` `{`⟨*description*⟩`}`

The workhorse of the affiliation database is the `\newaffil` command, which defines a new command that expands to call the revtex4-1 `\affiliation` command. For instance, `\newaffil{UFooBar}{Bar, UFoo}` defines a new command `\affilUFooBar` that expands to `\affiliation{Bar, UFoo}`.

So far, the following affiliation commands are provided:

| | |
|---|---|
| `\affilTODO` | Special, marks that an affiliation is not provided. |
| `\affilEQuSUSyd` | Centre for Engineered Quantum Systems, University of Sydney. |
| `\affilEQuSMacq` | Centre for Engineered Quantum Systems, Macquarie University. |
| `\affilUSydPhys` | School of Physics, University of Sydney. |
| `\affilIQC` | Institute for Quantum Computing, University of Waterloo. |
| `\affilUWPhys` | Department of Physics, University of Waterloo. |
| `\affilUWAMath` | Department of Applied Mathematics, University of Waterloo. |
| `\affilUWChem` | Department of Chemistry, University of Waterloo. |
| `\affilPI` | Perimeter Institute for Theoretical Physics. |
| `\affilCIFAR` | Canadian Institute for Advanced Research. |
| `\affilCQuIC` | Center for Quantum Information and Control, University of New Mexico. |
| `\affilIBMTJW` | IBM T. J. Watson Research Center. |

### 2.2.4 Internal-Use Commands

`\booloption`
`\newnew`

These commands are used internally by revquantum to define new boolean package options, and to declare new kinds of `\newcommand` commands. For example, `\newnew` is used to define `\newoperator`, which in turn defines new commands for named mathematical operators (e.g. Tr).

`\sectionautorefname`
`\algorithmautorefname`
`\equationautorefname`
`\lemmaautorefname`

These commands configure hyperref's `autoref` command for use with revtex4-1, so that `autoref` correctly describes Section, Algorithm, and Lemma, and also follows the standard notation for equation references.

`\boolean@xetex`

This boolean variable uses iftex to check if XeTeX is installed. If iftex is not available, then we assume plain LaTeX.

## 2.3 New Colors

The revquantum package also provides a color palette that is safe for colorblind readers and for printing, the Color Universal Design palette of Okabe and Ito.

| | |
|---|---|
| `cud-black` | ● |
| `cud-orange` | ● |
| `cud-sky-blue` | ● |
| `cud-bluish-green` | ● |
| `cud-yellow` | ● |
| `cud-blue` | ● |
| `cud-vermillion` | ● |
| `cud-reddish-purple` | ● |

These colors are defined as xcolor color names, such that they can be used in all packages which depend on xcolor. In particular, CUD colors can be directly used from tikz. To use with tikz, however, revquantum must be loaded *after* tikz.

# 3 Implementation

```
1
2 \usepackage{ifthen}
3
```

## 3.1 XeTeX Detection

We make a new boolean variable to track if XeTeX is being used.

```
4
5 \newboolean{@xetex}
6 \setboolean{@xetex}{false}
7 \IfFileExists{iftex.sty}{
8     \wlog{[INFO] iftex loaded}
9     \usepackage{iftex}
10 }{
11     \newif\ifXeTeX
12     \XeTeXfalse
13 }
14 \ifXeTeX
15     \wlog{[INFO] Using XeTeX}
16     \setboolean{@xetex}{true}
17 \else
18     \setboolean{@xetex}{false}
19 \fi
20
```

## 3.2 Notation

\newnew

```
21 \newcommand{\newnew}[2]{
22     \expandafter\newcommand\csname new#1\endcsname[1]{
23       \expandafter\newcommand\csname ##1\endcsname{#2{##1}}
24     }
25 }

26 \newnew{operator}{\operatorname}
27 \newnew{rm}{\mathrm}
28
29 \newoperator{Tr}
30 \newoperator{Cov}
31 \newoperator{supp}
32 \newoperator{diag}
33 \newoperator{rank}
34
```

```
\ii
35 \newcommand{\ii}{\mathrm{i}} % Outside what newnew currently supports.
\dd
36 \newcommand{\dd}{\mathrm{d}} % Outside what newnew currently supports.
37
38 \newrm{e}
39 \newrm{TVD}
40 \newrm{T}
41
\defeq
42 \newcommand{\defeq}{\mathrel{:=}}
\expect
43 \newcommand{\expect}{\mathbb{E}}
\id
44 \newcommand{\id}{\openone}
45
```

We want to only conditionally use `{stmaryrd}` if it's available, and otherwise, hack up a few commands from that package.

```
46  \IfFileExists{stmaryrd.sty}{
47      \RequirePackage{stmaryrd}
48  }{
49      \PackageWarning{revquantum}{
50          The stmaryrd package is not available,
51          so some commands (e.g.: double-brackets) will look wrong.
52      }
53      \newcommand{\llbracket}{[\!\!\hspace{1.5pt}[}
54      \newcommand{\rrbracket}{]\!\!\hspace{1.5pt}]}
55  }
```

## 3.3 Options Handling

We will need to define a few options to make the package nicer to use. We do so by making a new command, `\booloption {`⟨*boolname*⟩`} {`⟨*optionname*⟩`} {`⟨*default*⟩`}`.

```
\booloption
56 \newcommand{\booloption}[3]{
57     \newboolean{#1}
58     \setboolean{#1}{#3}
59     \ifthenelse{\equal{#3}{true}}{
```

Default is true, so we need a "no" option to turn off the new boolean.

```
        \DeclareOption{no#2}{\setboolean{#1}{false}}
    }{
```

Default is false, so we need an option to turn on the new boolean.

```
        \DeclareOption{#2}{\setboolean{#1}{true}}
    }
}
```

We then use this new command to define the options for revquantum, `final`, `pretty`, `uselistings`, `nobibtexhacks`, and `strict`.

```

\booloption{@final}{final}{false}
\booloption{@pretty}{pretty}{false}
\booloption{@uselistings}{uselistings}{false}
\booloption{@bibtexhacks}{bibtexhacks}{true}
\booloption{@strict}{strict}{false}

\ProcessOptions\relax

```

For the `strict` option, we do one last thing and define a new macro that either raises a warning or an option depending on whether `strict` has been set as an option.

`\rq@quasiwarn`

```
\ifthenelse{\boolean{@strict}}{
    \newcommand{\rq@quasiwarn}{
        \PackageError{revquantum}
    }
}{
    \newcommand{\rq@quasiwarn}{
        \PackageWarning{revquantum}
    }
}
```

## 3.4 Unforgivable BibTeX Hacks

These hacks include the title of each reference in the BibTeX output by redefining the part of revtex4-1 on the fly which is responsible for writing out the bibdata. Note that these hacks *must* come before importing packages, or else revtex4-1 will have already written out its control notes.

```

\ifthenelse{\boolean{@bibtexhacks}}{\def\@bibdataout@aps{%
 \immediate\write\@bibdataout{%
  @CONTROL{%
   apsrev41Control,author="08",editor="1",pages="0",title="0",year="1",eprint="1"%
  }%
 }%
```

```
\if@filesw
 \immediate\write\@auxout{\string\citation{apsrev41Control}}%
\fi
}}{}

```

## 3.5 Imports

Here, we import a few other useful packages and configure them according to the options passed by the user. In handling the fonts specified by `[pretty]`, we must be careful to do so in a way that is supported by XeTeX. Note that we only load color if neither tikz nor xcolor have already been imported, and if we are not using listings. In the latter case, we will load xcolor instead so that we can make listings play nicer with our own custom palette. Also of note is that we do not import hyperref yet, as it must go last to avoid duplicating reference names.

```

\RequirePackage{amsmath}
\RequirePackage{amsfonts}
\RequirePackage{amsthm}
\RequirePackage{amssymb}
\RequirePackage{amsbsy}
\@ifpackageloaded{tikz}{}{%
    \@ifpackageloaded{xcolor}{}{%
        \ifthenelse{\boolean{@uselistings}}{}{%
            \RequirePackage[usenames,dvipsnames]{color}%
        }%
    }%
    \AtBeginDocument{%
        \@ifpackageloaded{tikz}{%
            \rq@quasiwarn{tikz loaded, but after revquantum. This may not work.}%
        }%
    }%
}
\RequirePackage{braket}
\RequirePackage{graphicx}
\RequirePackage[english]{babel}
\ifthenelse{\boolean{@pretty}}{
    \ifthenelse{\boolean{@xetex}}{
        % http://tex.stackexchange.com/a/50593
        \usepackage{fontspec}
        \usepackage{mathpazo}
        \setmainfont
         [ BoldFont       = texgyrepagella-bold.otf ,
           ItalicFont     = texgyrepagella-italic.otf ,
           BoldItalicFont = texgyrepagella-bolditalic.otf ]
         {texgyrepagella-regular.otf}
    }{
        \RequirePackage{mathpazo}
```

```
    }
}{}
\ifthenelse{\boolean{@uselistings}}{
    \RequirePackage{xcolor}
    \RequirePackage{listings}
    \RequirePackage{textcomp} % Make sure we have a ‘ for writing Mathematica.
}{}
\ifthenelse{\boolean{@bibtexhacks}}{
    \RequirePackage{letltxmacro}
    \RequirePackage{etoolbox}
}{}
```

### 3.5.1 Theorem Environments

```

\newtheorem{theorem}{Theorem}
\newtheorem{lemma}{Lemma}

```

### 3.5.2 algorithm and algpseudocode Setup

```

\RequirePackage{algorithm}
\RequirePackage{algpseudocode}
    \renewcommand{\algorithmicrequire}{\textbf{Input:}}
    \renewcommand{\algorithmicensure}{\textbf{Output:}}
    \newcommand{\inlinecomment}[1]{\Comment {\footnotesize #1} \normalsize}
    \newcommand{\linecomment}[1]{\State \(\triangleright\) {\footnotesize #1} \normalsize}

```

### 3.5.3 listings Setup

Here, we provide special support for scientific languages like Python and Mathematica, as well as for legacy environments. This support consists of configuring escapes, quoting, providing additional keywords, etc.

```

\ifthenelse{\boolean{@uselistings}}{
    \definecolor{comment-color}{gray}{0.5}

    \lstset{
        basicstyle=\footnotesize,
        commentstyle=\color{comment-color},
        frame=lines,
        gobble=4,
        numbers=left,
        numberstyle=\tiny, stepnumber=5,
        numbersep=5pt,
        keywordstyle=\color{cud-bluish-green!85!black}\bfseries,
        keywordstyle={[2]\color{cud-sky-blue!75!black}},
        emphstyle=\color{cud-vermillion}
    }
```

```
    \ifthenelse{\boolean{@xetex}}{
        \RequirePackage{sourcecodepro}
        \lstset{basicstyle=\footnotesize\sourcecodepro}
    }{}

    \lstdefinestyle{matlab}{
        language=MATLAB,
        mathescape=true
    }

    \lstdefinestyle{python}{
        language=Python,
        mathescape=true,
        showstringspaces=false,
        morekeywords={as,async,await}
    }

    \lstdefinestyle{mathematica}{
        language=Mathematica,
        upquote=true, % Needed to deal with the context symbol `.'
        literate=
            {->}{$\to$}1
            {!=}{$\neq$}1
            {\\[DoubleStruckOne]}{${\id}$}1
            {\\[Sigma]}{$\sigma$}1
            {(x)}{$\otimes$}1 % CG: I have the distinct impression this will break. Badly.
    }
}{}

```

## 3.6 Import Warnings

The following command will cause a warning to be emitted if the package named by its argument is loaded. To make robust against the order in which packages are loaded, all such logic happens at `\begin{document}`. This code is adapted from the solution provided by Martin Scharrer at http://tex.stackexchange.com/a/16200/615.

`\rq@warnpackage`

```
\newcommand{\rq@warnpackage}[1]{
    \AtBeginDocument{%
        \@ifpackageloaded{#1}{%
            \rq@quasiwarn{The #1 package is known to be incompatible with revtex4-1. You may en
        }{}
    }
}
```

With this command in place, we can now issue specific warnings for individual "bad" packages.

```
\rq@warnpackage{subcaption}
```

## 3.7 Slightly More Forgivable BibTeX Hacks

Next, we include a solution suggested by egreg for a rather annoying `{revtex4-1}` bug. In particular, we will set up `language={en}` as an alias for `language={english}`, so that `{revtex4-1}` will no longer raise `{babel}` errors for the undefined language. As with our unforgivable hacks, we will guard the forgivable hacks with the `[nobibtexhacks]` option.

`\ORIGselectlanguage`

```
\ifthenelse{\boolean{@bibtexhacks}}{
    \LetLtxMacro{\ORIGselectlanguage}{\selectlanguage}
    \DeclareRobustCommand{\selectlanguage}[1]{%
      \@ifundefined{alias@\string#1}
        {\ORIGselectlanguage{#1}}
        {\begingroup\edef\x{\endgroup
           \noexpand\ORIGselectlanguage{\@nameuse{alias@#1}}}\x}%
    }
}{}

```

`\definelanguagealias`

```

\newcommand{\definelanguagealias}[2]{%
  \@namedef{alias@#1}{#2}%
}


\definelanguagealias{en}{english}
\definelanguagealias{EN}{english}
\definelanguagealias{English}{english}

```

We will also redefine `\doibase` to eat any newlines following it, and to prefer HTTPS to HTTP. This will fix a rather annoying bug where line breaks in the *.bbl source can introduce extraneous spaces into the target of each citation's `\href`.

`\doibase`

```

\ifthenelse{\boolean{@bibtexhacks}}{
    % Ensure that the \doibase command is defined, just in case.
    \providecommand{\doibase}{}
    \renewcommand{\doibase}[1]{https://dx.doi.org/\ifdefempty{#1}{}{#1}}
}{}
```

```

%    \end{macrocode}
% \end{macro}


% \subsection{TODO Support} %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% These commands provide a way of marking items as needing to be done before
% the final version (denoted by the |final| package option).

% \begin{macro}{\todo}
%    \begin{macrocode}

\ifthenelse{\boolean{@final}}{
    \newcommand{\todo}[1]{%
        \PackageError{revquantum}{Unaddressed TODO}%
        \rq@todo{#1}%
    }
}{
    \newcommand{\todo}[1]{%
        \PackageWarning{revquantum}{Unaddressed TODO}%
        \rq@todo{#1}%
    }
}

```

We also define a `\citeneed` command for the special case of a missing citation. As per Steve Flammia's suggestion, this is formatted in analogy to the infamous Wikipedia annotation.

`\citeneed`

```
\ifthenelse{\boolean{@final}}{
    \newcommand{\citeneed}{%
        \PackageError{revquantum}{citation needed}%
        \rq@todo{[citation needed]}%
    }
}{
    \newcommand{\citeneed}{%
        \PackageWarning{revquantum}{citation needed}%
        \rq@todo{[citation needed]}%
    }
}
```

Both of these macros are based on the `\rq@todo` macro, which performs the formatting for TODOs.

`\rq@todo`

```
\newcommand{\rq@todo}[1]{%
    {\color{magenta} #1}%
}
```

We also provide a few other special cases below.

\TODO

```
\newcommand{\TODO}{\todo{TODO}}
```

\todolist

```
\newcommand{\todolist}[1]{\todo{
    \begin{itemize}
        #1
    \end{itemize}
}}

```

## 3.8 Color Universal Design

```
\definecolor{cud-black}          {RGB}{0,0,0}
\definecolor{cud-orange}         {RGB}{230,159,0}
\definecolor{cud-sky-blue}       {RGB}{86,180,233}
\definecolor{cud-bluish-green}   {RGB}{0,158,115}
\definecolor{cud-yellow}         {RGB}{240,228,66}
\definecolor{cud-blue}           {RGB}{0,114,178}
\definecolor{cud-vermillion}     {RGB}{213,94,0}
\definecolor{cud-reddish-purple}{RGB}{204,121,167}
```

## 3.9 Affiliation Database

\newaffil

```
\newcommand{\newaffil}[2]{
    \expandafter\newcommand\csname affil#1\endcsname{
        \affiliation{
            #2
        }
    }
}
```

### 3.9.1 General Affiliations

```

\newaffil{TODO}{\TODO}

```

### 3.9.2 Australia

```

\newaffil{EQuSUSyd}{
    Centre for Engineered Quantum Systems,
    University of Sydney,
    Sydney, NSW, Australia
}
\newaffil{EQuSMacq}{
```

```
    Centre for Engineered Quantum Systems,
    Macquarie University,
    Sydney, NSW, Australia
}
\newaffil{USydPhys}{
    School of Physics,
    University of Sydney,
    Sydney, NSW, Australia
}

```

### 3.9.3 Canada

```

\newaffil{IQC}{
    Institute for Quantum Computing,
    University of Waterloo,
    Waterloo, ON, Canada
}
\newaffil{UWPhys}{
    Department of Physics,
    University of Waterloo,
    Waterloo, ON, Canada
}
\newaffil{UWAMath}{
    Department of Applied Mathematics,
    University of Waterloo,
    Waterloo, ON, Canada
}
\newaffil{UWChem}{
    Department of Chemistry,
    University of Waterloo,
    Waterloo, ON, Canada
}
\newaffil{PI}{
    Perimeter Institute for Theoretical Physics,
    31 Caroline St. N,
    Waterloo, Ontario, Canada N2L 2Y5
}
\newaffil{CIFAR}{
    Canadian Institute for Advanced Research,
    Toronto, ON, Canada
}

```

### 3.9.4 United States

```

\newaffil{CQuIC}{
    Center for Quantum Information and Control,
    University of New Mexico,
    Albuquerque, NM 87131-0001, USA
```

```
}
\newaffil{IBMTJW}{
    IBM T. J. Watson Research Center,
    Yorktown Heights, New York 10598, USA
}
```

## 3.10 hyperref Setup

Finally, we load hyperref separately so that it can go last.

Get rid of hyperref's ugly boxes. From:http://tex.stackexchange.com/a/51349

```
\RequirePackage[breaklinks=true]{hyperref}

\hypersetup{
  colorlinks   = true, %Colours links instead of ugly boxes
  urlcolor     = blue, %Colour for external hyperlinks
  linkcolor    = blue, %Colour of internal links
  citecolor    = red %Colour of citations
}
```

### 3.10.1 autoref Setup

We must declare our autoref configuration at the beginning of the document to keep other packages from clobbering it.

\sectionautorefname

```
\AtBeginDocument{%
    \def\sectionautorefname{Section}%
}
```

\algorithmautorefname

```
\AtBeginDocument{%
    \def\algorithmautorefname{Algorithm}%
}
```

\equationautorefname See http://tex.stackexchange.com/a/66150.

```
\AtBeginDocument{%
    \def\equationautorefname~#1\null{(#1)\null}%
}
```

\lemmaautorefname

```
\AtBeginDocument{%
    \newcommand{\lemmaautorefname}{Lemma}%
}
```

# Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.