Yiannis Lazarides

The phd-pkgmanager Package

LaTeX Document Processing

Book Design Monographs

Camel Press

phd-pkgmanager package
Design of Ancillary Book Elements

Dr Yiannis Lazarides

University of the Witwatersrand

Camel Press
P.O.Box 40259
Larnaca
Cyprus

Cover image: The cover image shows Jo Bodeon, a back-roper in the mule room at Chace Cotton Mill. Burlington, Vermont. This and other similar images in this book were taken by Lewis W. Hine, in the period between 1908-1912. These images as well as social campaigns by many including Hine, helped to formulate America's anti-child labour laws.

Copyright ©2015–2024 Dr Yiannis Lazarides.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, version 1.2, with no invariant sections, no front-cover texts, and no back-cover texts.

A copy of the license is included in the appendix.

This document is distributed in the hope that it will be useful, but without any warranty; without even the implied warranty of merchantability or fitness for a particular purpose.

Contents

Part I

USER MANUAL

1 Documentation

The purpose of this package is to assist in managing the loading of a set of useful packages and minimize the preamble of a LaTeX document. I developed the package while witing a series of books.

2 How to use the package

```
\documentclass[book,11pt,oneside,openany]{phddocx}
\usepackage[essential,math,math+,tikz]{phd-pkgmanager}
```

```
\documentclass[book,10pt,oneside,openany]{phddocx}
\usepackage[essential,math,math+,tikz]{phd-pkgmanager}
```

Packages have been grouped into classes and loaded using the ⟨*options*⟩. The sets of packages are then loaded in correct order and in addition if there is a clash correct for the clash.

```
\documentclass[book,10pt,oneside,openany]{phddocx}
% Geometry settings
\usepackage{geometry}
% General packages settings
\usepackage[essential,math,math+,tikz,biblatex,
   hyper,algos]{phd-pkgmanager}
% fonts
\addbibresource{jobname.bib}
\hypersetup{...}
\makeindex
\begin{document}
 ...
\end{document}
```

It is also possible to use higher level settings, that would load recommended packages based on *topics*.

```
\usepackage[topic=computing]{phd-pkgmanager}
```

Some of the topics available are computing,latex for notes or books on LaTeX topics, mathematics, science, linguistics.

At this point is good to pause and discuss as to what the class should set. In my opinion the class is responsible to set the looks of the document, so the package manager does not provide any styling packages, such as enumitem etc. More about this later.

2.1 Options

The package is made upof smaller packages, classifying the packages into the following categories: essential, fonts, typography, math, ancient,programming,tikz,chemistry,miscellaneous. Each category loads one or more packages, with what I found over the years as reasonable package options and used by the monographs I published.

	package	category
1	array	core
2	longtable	core
3	threeparttable	core
	Figures related	
4	graphicx	core
5	wrapfig	core
6	grfext	core
7	rotating	core
	Table related	
8	booktabs	core
9	tabularx	core
10	array	core
11	longtable	core
12	multirow	core
13	colortbl	core
14	threeparttable	core
15	diagbox	core
	Text and general	
16	acronym	core
17	pdfpages	core
18	comment	core
19	needspace	core

The option math loads a basic set for maths. The packages loaded depend on the mathunicode option. The extramath loads a few extra packages that are popular for arrays and theorems.

`all` =
`core` *essential packages*
`core` *essential packages*

Part II

IMPLEMENTATION

3 Implementation Strategy

The implementation is divided into parts. Perhaps cutting, these parts into smaller packages might have been a better choice, but as the aim of the package is to minimize the loading of packages and let `phd` to handle this, it made more sense to me, anyway to keep everything together.

The Package Manager This section is responsible for pre-loading packages, resolving conflicts and providing all interfacing commands.

The Sectioning Layouts Manager This section manages the design of complex layouts for sectioning commands.

The Image Page Manager This section manages the design of pages that consist primarily of images and complex page layouts.

Common Macros We provide a number of predefined commands for macros that us and other people found useful.

MWE The package generates a large number of Minimum Working Examples that we use for testing. Most of them can also used as examples for training or self-study.

4 Preliminaries

The basic requirement for the Package Manager is to load an adequate number of packages to enable the typesetting of a diverse number of large documents without requiring additional packages to be loaded by typical groups of authors. This has its advantages, but of course it does slow things down. A long term objective is to select packages depending as an option on the type of document being prepared.

```
1 ⟨*package⟩
2 ⟨@@=phd⟩
```

Standard file identification. We first announce the package and require that it be used with LaTeX2e.

```
3 ⟨/package⟩
4 ⟨*package|essential|tikz|chemistry|ancient|typography|math|programming⟩
5 \NeedsTeXFormat{LaTeX2e}[2020/02/02]
6 ⟨/package|essential|tikz|chemistry|ancient|typography|math|programming⟩
7 ⟨*package⟩
8 \ProvidesPackage{phd-pkgmanager}[2020/1/13 v1.0 less preamble (YL)]%
9 \ExplSyntaxOn
10 \cs_gset:Npn \MakePrivateLetters
11   {
12     \char_set_catcode_letter:N \@
13     \char_set_catcode_letter:N \_
14     \char_set_catcode_letter:N \:
15   }
16   \MakePrivateLetters
17 \def\FIRE{Fire}
```

5 Utilities

In order to keep track of all the packages and keys we require a number of macros will be defined first.

Each of the packages used by this document is loaded conditionally. However, it might be nice to know if we have a complete set. So we define `\ifcomplete` which starts true, but gets set to false if any package is missing. Some code is necessary in order to manage the complexity.

I am indebted to the source of **symbols.tex** for the ideas and structure of some of the macros, which mostly I converted to LaTeX3 syntax.

There are a number of symbols (e.g., \Square) that are defined by multiple packages. In order to typeset all the variants in this document, we have to give glyph a unique name. To do that, we define :

\savesymbol{⟨*XXX*⟩}, which renames a symbol from \XXX to \origXXX, and \restore_symbol:{⟨*yyy*⟩}{⟨*XXX*⟩} which renames \origXXX back to \XXX and defines a new command, \yyyXXX, which corresponds to the most recently loaded version of \XXX.

This implementation of \cs{save_symbol:} and \cs{restore_symbol:} was based on the savesym package, which started with symbol.tex's old definitions of those macros and improved upon them. However, \renamerobustsymbol and \ifnotsavedsym are from the list of symbols documentation.

\g_phd_packages_loaded_clist} {\marg{clist} Holds a list of all packages loaded by the phd package. \g_phd_packages_not_found:n}{\marg{clist} Holds a list of all packages not found.

These are really long names, but I want to follow the LaTeX3 Teams' suggestions and recommendations.

save_symbol: ⟨*symbol name*⟩ An explorified version of savesymbol. In the old style the original command was set to relax, this caused errors and I set it to undefine. The joys of TEXprogramming!!!!! typeset the first argument the macro names

```
%\cs_new:Npn \save_symbol: #1
%  {
%    \cs_gset_eq:cc {orig#1} {#1}
%    \cs_undefine:c {#1}
%  }
%
```

\test_also_underscore:Nnn {⟨*x*⟩},{⟨*y*⟩} ..... `a b some llong word to see what happens`

Just testing, something is weird here.

\test_also_underscore:Nnn {⟨*x*⟩},{⟨*y*⟩} .....

Just testing, something is weird here.

\savesymbol {⟨*symbol name*⟩} ..... `pkg`

An alias for save_symbol:.

```
\cs_set_eq:NN \savesymbol\save_symbol:
\ExplSyntaxOff
```

\restore_symbol: {⟨*symbol prefix*⟩} {⟨*symbol name*⟩}

Restore a previously saved symbol, and rename the current one.

```
\ExplSyntaxOn
\cs_new:Npn \restore_symbol: #1 #2
  {
    \cs_gset_eq:cc {#1#2} {#2}
    \cs_gset_eq:cc {#2} {orig#2}
  }
\ExplSyntaxOff
```

Rename a robust command.

```
\newcommand*{\renamerobustsymbol}[2]{%
  \expandafter\let\expandafter\origrealcommand
    \csname #2\space\endcsname
    \expandafter\global\expandafter\let\csname#1#2\endcsname=\origrealcommand
}
```

Test if a symbol is not saved.

```
\def\ifnotsavedsym@helper#1#2!{\expandafter\ifx\csname orig#2\endcsname\relax}
\newcommand*{\ifnotsavedsym}[1]{%
  \expandafter\ifnotsavedsym@helper\string#1!%
}

\newif\ifcomplete
```

For debugging purposes we define a switch that enables us to toggle on and off the loading of packages.

```
\ExplSyntaxOn
\clist_new:N \g_phd_packages_loaded_clist:n
\clist_new:N \g_phd_packages_not_found:n
\newif\ifloadpackages
\loadpackagestrue

\newcommand{\missingpkgs}{}
\newcommand{\foundpkgs}{}

```

\ifstylefileexists {⟨*true code*⟩}{⟨*false code*⟩} .....

Checks if a `.sty` file exists. **\ifstylefileexists** is just like LaTeXe's **\IfFileExists**, except that it appends `.sty` to its first argument. `\ifstylefileexists` is the same as `\ifstylefileexists*`, but it additionally adds its first argument to a list (`\missingpkgs`) and marks the document as incomplete (with `\completefalse`) if the `.sty` file doesn't exist.

```
\NewDocumentCommand\ifstylefileexists {s m m m } {
  \IfBooleanTF #1
    {\ifstylefileexists_star {#2}{#3}{#4} }
    {\ifstylefileexists_aux {#2}{#3}{#4}}
}
```

Next define the auxiliaries.

```
\cs_new:Npn \ifstylefileexists_star #1 #2 #3 {
\ifloadpackages
\file_if_exist:nTF {#1}
  {
    \exp_after:w \ifx\csname ver@#1.sty\endcsname\relax
            \PackageInfo{phd-pkgmanager}{package~#1~loaded.}
    \else
      \PackageInfo{phd-pkgmanager}{package~#1~already~loaded.}
    \fi
     \clist_gput_right:Nn \g_phd_packages_loaded_clist:n {#1}
```

```
    #2
  }
  {
    #3
    \clist_gput_right:Nn \g_phd_packages_not_found:n {#1}
    \PackageInfo{phd-pkgmanager}{package~#1~not~found.}
  }
\fi
}

\cs_new:Npn \ifstylefileexists_aux #1 #2 #3 {

\file_if_exist:nTF {#1.sty}
  {

  \if_meaning:w \ver@#1.sty\relax

       \clist_gput_right:Nn \g_phd_packages_loaded_clist:n {#1}
     \PackageInfo{phd-pkgmanager}{package~#1~not~loaded.}

     \else:

     \PackageInfo{phd-pkgmanager}{package~#1~already~loaded.}
      #2
    \fi:
  }
  {
   #3
    \clist_gput_right:Nn \g_phd_packages_not_found:n {#1}
    \PackageInfo{phd-pkgmanager}{package~#1~not~found.}
  }

}
```

\LoadPackageAll [⟨*package options*⟩] {⟨*package name*⟩}.....

Checks if a `.sty` file exists. Loads it with all options for all engines and or bundles.

```
\NewDocumentCommand\LoadPackageAll { o m } {
  \bool_if_exist:cTF{#2_bool}
  {  }
  {\bool_new:c {#2_bool}
   \cs_new:cpn {#2_name} {\pkg{#2}}
   \ifstylefileexists{#2}
       {\bool_gset_true:c {#2_bool}
         \IfValueTF
           {\PassOptionsToPackage
            \RequirePackage{#2}
            }{\RequirePackage{#2}}
       }
       {error cannot be loaded
```

```
        \bool_gset_false:c {#2_bool}
      }
  }
}
\ExplSyntaxOff
```

To find out if a package has already been loaded, use **\@ifpackageloaded**{⟨*package*⟩}{⟨*true*⟩}{⟨*false*⟩}.
\@ifpackagelater To find out if a package has already been loaded with a version more recent
\@ifclasslater than version, use \@ifpackagelater⟨*hpackagei*⟩⟨*version*⟩⟨*true*⟩⟨*false*⟩. \@ifpackagewith
To find out if a package has already been loaded with at least the options options, use \@ifpackagewith⟨*package*⟩⟨*opt*

There exists one package that can't be tested with the above commands: the fontenc package pretends that it was never loaded to allow for repeated reloading with different options (see ltoutenc.dtx for details).

6 Utility macros for displaying symbols and fonts

In the sections that follow, we use a number of utilities for displaying fonts and utilities in tables and figures, we collect them here and make them available to the user for document use. Many are modifications from other packages.

```
% From stmarysrd symbols package
% A very convenient command to typeset symbols.
% Much preferable than tables. Slight modifications to
% make it a bit more clear
% CHECK END SYMBOLS
\newcommand\Symbols{\flushleft}
\def\endSymbols{\endflushleft}
\def\dosymbol#1{%
   \leavevmode\hbox to .33\textwidth{%
    \hbox to 1.2em%
    {\hss$#1$\hfil}%
   \footnotesize\texttt{\string#1}\hss}%
   \penalty10}
```

7 Key Definitions

```
\ExplSyntaxOn
\clist_set:Nn \l_tmpa_clist{all,essential,math,typography,programming,tikz,chemistry,physics}
```

The **\noload_clist** holds packages that should not be loaded. Is settable through the key-value interface nd is initially empty.

```
\clist_new:N \noload_clist
\clist_set:Nn \noload_clist{}
\clist_map_inline:Nn\l_tmpa_clist{\bool_new:c {__phd_#1_bool}}
\bool_new:N\__phd_explplus_bool %extra expl packages


\DeclareKeys[phd/pkgm]
 {
```

```
   all .bool_set:N             = \__phd_all_bool,
   all .default:n                = true,
   essential .bool_set:N       = \__phd_essential_bool,
   essential.default:n         = true,
   math .bool_set:N            = \__phd_math_bool,
   math .default:n             = true,
   typography .bool_set:N      = \__phd_typography_bool,
   typography .default:n       = true,
   programming .bool_set:N     = \__phd_programming_bool,
   programming .default:n      = true,
   tikz .bool_set:N            = \__phd_tikz_bool,
   tikz .default:n             = true,
   chemistry .bool_set:N       = \__phd_chemistry_bool,
   chemistry .default:n        = true,
   physics .bool_set:N         = \__phd_physics_bool,
   physics .default:n          = true,
   ancient .bool_set:N       = \__phd_ancient_bool,
   ancient .default:n        = true,
   expl+ .bool_set:N         = \__phd_explplus_bool,
   expl+ .default:n          = true,
   thesis .meta:nn         = {phd/pkgm}{chemistry=true},
   exclude .code            = \clist_put_right:Nn\noload_clist{#1},
  }


\keys_set:nn{phd/pkgm}
  {
    all=false,ancient=false,chemistry=false,essential,math,typography,programming,expl+
  }

\ProcessKeyOptions[phd/pkgm]
\bool_if:NTF\__phd_all_bool
  {
 \clist_map_inline:nn {phd-essential,phd-math,phd-typography,phd-programming,phd-tikz,phd-
chemistry,phd-physics, phd-ancient,xparse,xtemplate,xcoffins,l3benchmark}{\RequirePackage{#1}}
  }
  {
 \bool_if:NT\__phd_essential_bool{\RequirePackage{phd-essential}}
 \bool_if:NT\__phd_math_bool{\RequirePackage{phd-math}}
 \bool_if:NT\__phd_typography_bool{\RequirePackage{phd-typography}}
 \bool_if:NT\__phd_programming_bool{ \RequirePackage{phd-programming} }
 \bool_if:NT\__phd_tikz_bool{\RequirePackage{phd-tikz}}
 \bool_if:NT\__phd_chemistry_bool{\RequirePackage{phd-chemistry}}
 \bool_if:NT\__phd_physics_bool{\RequirePackage{phd-physics}}
 \bool_if:NT\__phd_ancient_bool{\RequirePackage{phd-ancient}}
 \bool_if:NT \__phd_explplus_bool{\RequirePackage{xparse,xtemplate,xcoffins}}
  }
```

Before we load the individual packages, we check if they have been excluded.

```
 %\clist_show:N\noload_clist
```

```
  \clist_map_inline:Nn\noload_clist{
    \clist_remove_all:Nn\core_packages_clist{#1}
  }
  %\clist_show:N\core_packages_clist
  \clist_map_inline:Nn \core_packages_clist
    {
        \RequirePackage{#1}
    }
%
⟨/package⟩
```

8 Essential Packages

The internal package `phd-essential` provides a list of packages which I found essential for a reasonably long book. This includes packages for tables **booktabs**,Fear (2005) tabularx, longtable, multirow, array,colortbl, phd-lorems, lipsum, kantlipsum, blindtext, xspace, comment

8.1 Graphics

graphicx wrapfig, rotating, caption, pdfscape

```
⟨*essential⟩
\ProvidesExplPackage{phd-essential}{20/11/2023}{version1.0}{core libraries (YL)}
\clist_new:N\core_packages_clist
\clist_set:Nn \core_packages_clist{booktabs,tabularx,longtable,multirow,array,colortbl,threepar
\clist_put_right:Nn\core_packages_clist{lipsum,phd-lorems,kantlipsum,blindtext}
\clist_put_right:Nn \core_packages_clist{calc,xspace,comment}
\clist_put_right:Nn \core_packages_clist{graphicx,wrapfig,rotating,caption,subcaption,pdflscape
of,phd-epigraphs,varwidth,pifont,marvosym}
\clist_put_right:Nn \core_packages_clist{framed}
%\clist_map_inline:Nn \core_packages_clist{\RequirePackage{#1}}
 \@ifundefined{c@step}{\newcounter{step}}{}
 \newcommand\resetinc{\setcounter{step}{0}}
 \newcommand\inc{\stepcounter{step}\thestep}
% \RequirePackage{grfext}
% \DeclareGraphicsExtensions{.jpg, .JPG, .jpeg, .JPEG, .eps, .pdf, .PDF, .png, .PNG}
% \graphicspath{ {./images//} {./images-01//} {./graphics/}   {./images/cape//} {./images/rsa//
   %\AppendGraphicsExtensions{.png}
 %\PrintGraphicsExtensions
%\PassOptionsToPackage{quiet}{rotating}
%\RequirePackage{rotating}
\RequirePackage{ragged2e}
\RequirePackage{pict2e}
\RequirePackage{picture}
\PassOptionsToPackage{final}{pdfpages} %review the options
 \PassOptionsToPackage{smaller,printonlyused,withpage}{acronym}
    \RequirePackage{acronym}[2015/03/21]
    %\RequirePackage{phd-abbreviations}
     \RequirePackage{siunitx}
```

```
  \sisetup{fixed-exponent =0,
           scientific-notation = false}
 \PassOptionsToPackage{np}{numprint}%
 \RequirePackage{numprint}
 \RequirePackage[super]{nth}
⟨/essential⟩
```

9 Programming

Package that are commonly used by programmers. environ, etoolbox parselines, upquote, alphalph.

```
⟨*programming⟩
\ProvidesExplPackage{phd-programming}{2023/11/20}{version1}{Core packages programming}
\RequirePackage{etoolbox}
\RequirePackage{environ}
\RequirePackage{parselines}
\RequirePackage{upquote}
\RequirePackage{alphalph}
⟨/programming⟩
```

10 Tikz preloads

```
⟨*tikz⟩
\ProvidesPackage{phd-tikz}[20/11/2023 version1[load tikz libraries]
\makeatletter
\RequirePackage{tikz}
\usetikzlibrary{%
  arrows, %
  calc,%
  fit,%
  patterns,%
  plotmarks,%
  shapes.geometric,%
  shapes.misc,%
  shapes.symbols,%
  shapes.arrows,%
  shapes.callouts,%
  shapes.multipart,%
  shapes.gates.logic.US,%
  shapes.gates.logic.IEC,%
  er,%
  automata,%
  backgrounds,%
  chains,%
  topaths,%
  trees,%
  petri,%
  mindmap,%
```

```
  matrix,%
  calendar,%
  folding,%
  fadings,%
  through,%
  positioning,%
  scopes,%
  decorations.fractals,%
  decorations.shapes,%
  decorations.text,%
  decorations.pathmorphing,%
  decorations.pathreplacing,%
  decorations.footprints,%
  decorations.markings,%
  shadows}
 \usetikzlibrary{tikzmark}
\usetikzlibrary{datavisualization}
\usetikzlibrary{datavisualization.formats.functions}
% pgfplots latest compatibility
 \RequirePackage{pgfplots}
 \pgfplotsset{compat=1.18}
  \RequirePackage{pgfplotstable}
 \RequirePackage{forest}
 \LoadPackageAll{drawstack}
 \usetikzlibrary{tikzmark}
⟨/tikz⟩
```

11 Typography package

This package loads the lettrine package etc.

```
⟨*typography⟩
\ProvidesExplPackage{phd-typography}{20/11/2023}{version1}{Core packages typography}
\RequirePackage{soul}
\sethlcolor{thehighlight}
\RequirePackage{lettrine}
\def\dropcap#1#2{\lettrine[lines=3, lraise=0.1, nindent=0em, slope=.1em]{#1}{#2}}
⟨/typography⟩
```

```
⟨*ancient⟩
\ProvidesExplPackage{phd-ancient}{20/11/2023}{version1}{Core packages ancient (YL)}
\ExplSyntaxOn
\cs_gset:Npn \MakePrivateLetters
  {
    \char_set_catcode_letter:N \@
    \char_set_catcode_letter:N \_
    \char_set_catcode_letter:N \:
  }
\RequirePackage{staves}
\LoadPackageAll{uncial}
```

```
\LoadPackageAll{lineara}
\LoadPackageAll{linearb}
\LoadPackageAll{cypriot}
\LoadPackageAll{sarabian}
\LoadPackageAll{oldprsn}
\RequirePackage{hieroglf}
\RequirePackage{ugarite}
\RequirePackage{epiolmec}
⟨/ancient⟩
⟨*math⟩
\ProvidesExplPackage{phd-math}{20/11/2023}{version1}{easy math setup (YL)}
\clist_new:N \math_packages_clist
\clist_set:Nn\math_packages_clist
  {amsmath,amssymb,amsthm,amsopn,amscd,mathtools,xfrac,nicefrac,braket,stackrel,empheq}
\clist_map_inline:Nn\math_packages_clist{\RequirePackage{#1}}
\setcounter{MaxMatrixCols}{20}
\newcommand*\widefbox[1]{\fbox{\hspace{1em}#1\hspace{1em}}}
⟨/math⟩
```

12 Chemistry packages

mchem is a very popular package for chemistry load it, if the option chemistry is set.

```
⟨*chemistry⟩
\ProvidesExplPackage{phd-chemistry}{20/11/2023}{version1}{easy chemistry setup (YL)}
\PassOptionsToPackage{version=4}{mhchem}
\RequirePackage{mhchem}
⟨/chemistry⟩
```

13 Physics and related packages

the old physics package has been problematic over the years, use physics2 and laod with limited modules `ab`, `a.braket`. Best define your own shortcuts to suit your specialty.

```
⟨*physics⟩
 \ProvidesExplPackage{phd-physics}{20/11/2023}{version1}{easy physics setup (YL)}
 % must come after amsmath it will load it if not present then works with modules.
⟨@@=⟩
 \makeatletter
 \RequirePackage{physics2}
 \usephysicsmodule{ab,ab.braket}
⟨/physics⟩
⟨*package⟩
\wlog{*******************************}
\wlog{  END PHD-PKGMANAGER}
\wlog{*******************************}
⟨/package⟩
```

References

Fear 2005

S. Fear, Publication quality tables in latex, 2005