The `xkeymask` package

Ramkumar Ramachandra

December 25, 2023

This package is an extension of the `xkeyval` package by Hendri Adriaens that offers macros for "masking" certain keys, so that they are not used in certain places. The package was originally written for the purposes of the bonak project, for hiding certain keys in certain places, and has been released for a wider audience. The idea is that one first imports the package with a desired "prefix" in the document preamble, and then creates masks use at certain points in the document. The mask can also be queried or cleared at some later point. The main user interface is provided by the `\appendmask`, `\clearmask`, and `\ifinmask` commands. The following example:

```
\documentclass{article}
\usepackage{amsmath}
\usepackage[prefix=bonak]{xkeymask}

\makeatletter
\define@cmdkey[bonak]{frame}{D}{(#1)}
\define@cmdkey[bonak]{layer}{D}{(#1)}
\define@cmdkey[bonak]{layer}{d}{(#1)}
\define@cmdkey[bonak]{painting}{D}{(#1)}
\define@cmdkey[bonak]{painting}{E}{(#1)}
\define@cmdkey[bonak]{painting}{d}{(#1)}
\makeatother

\newcommand{\prim}[6]{
  \ensuremath{\mathsf{{#1}}_{#2}^{#3,#4,#5}}
  \setkeys*[bonak]{#1}{#6}}

\newcommand{\framep}[5]{\prim{frame}{#1}{#2}{#3}{#4}{#5}}
\newcommand{\layer}[5]{\prim{layer}{#1}{#2}{#3}{#4}{#5}}
\newcommand{\painting}[5]{\prim{painting}{#1}{#2}{#3}{#4}{#5}}

\newcommand{\eqnarg}[3]{\ifinmask[bonak]{#1}[#2]{\{#2:#3\}}{(#2:#3)}}

\begin{document}
  With mask:
  \appendmask[bonak]{layer}[D]
  \appendmask[bonak]{painting}[D,E]
  \appendmask[bonak]{cohfiller}[c]
  \begin{align*}
    \eqnarg{cohlayer}{d}{\framep{m}{n}{p}{q}{D=D}} \\
    \eqnarg{cohlayer}{l}{\layer{m}{n}{p}{q}{D=D, d=d}} \\
    \eqnarg{cohfiller}{c}{\painting{m}{n}{p}{q}{D=D, E=E, d=d}} \\
  \end{align*}
  Without mask:
  \clearmask
  \begin{align*}
    \eqnarg{cohlayer}{d}{\framep{m}{n}{p}{q}{D=D}} \\
    \eqnarg{cohlayer}{l}{\layer{m}{n}{p}{q}{D=D, d=d}} \\
    \eqnarg{cohfiller}{c}{\painting{m}{n}{p}{q}{D=D, E=E, d=d}} \\
  \end{align*}
\end{document}
```

produces the following output:

With mask:

$$(d : \mathsf{frame}_m^{n,p,q}(D))$$
$$(l : \mathsf{layer}_m^{n,p,q}(d))$$
$$\{c : \mathsf{painting}_m^{n,p,q}(d)\}$$

Without mask:

$$(d : \mathsf{frame}_m^{n,p,q}(D))$$
$$(l : \mathsf{layer}_m^{n,p,q}(D)(d))$$
$$(c : \mathsf{painting}_m^{n,p,q}(D)(E)(d))$$

Full source code listing

```
% xkeymask package by Ramkumar Ramachandra.
%%
%% ----------------------------------------
%% Copyright (C) 2022-2023 Ramkumar Ramachandra
%% ----------------------------------------
%%
%% This work may be distributed and/or modified under the
%% conditions of the LaTeX Project Public License, either
%% version 1.3c of this license or (at your option)
%% any later version. The latest version of this license
%% can be found at:
%%   https://www.latex-project.org/lppl.txt
%% and version 1.3c or later is part of all distributions
%% of LaTeX version 2008/05/04 or later.
%%
%% This work has the LPPL maintenance status "maintained".
%%
%% This Current Maintainer is Ramkumar Ramachandra.
%%
%% This work consists of the file xkeymask.sty.
%%
%% The following files constitute the xkeymask bundle and
%% must be distributed as a whole:
%%   README, xkeymask.sty, xkeymask.tex, xkeymask.pdf.
%%
\RequirePackage{kvoptions, xkeyval}
\ProvidesPackage{xkeymask}[v1.0 An extension of xkeyval with a mask]

% Option processing
\newif\ifxkeymask@prefix
\DeclareStringOption{prefix}
\AddToKeyvalOption*{prefix}{\xkeymask@prefixtrue}
\ProcessKeyvalOptions*

\ifxkeymask@prefix
  % We patch xkeyval's \setkeys to use a mask to ignore keys
  \makeatletter
  % Our global mask
  \def\XKM@mask{}
```

```latex
% First we define \appendmask
\long\def\XKM@appendmask[#1]{
  \XKV@for@o\XKV@fams\XKV@tfam{
    \xdef\XKM@hdr{\XKV@prefix\XKV@tfam @}
    \XKV@for@o{#1}\XKV@tempa{
      % Remove spaces around key name
      \expandafter\KV@@sp@def\expandafter\XKV@tempa\expandafter{\XKV@tempa}
      \XKV@addtolist@x\XKM@mask{\XKM@hdr\XKV@tempa}
    }
  }
}
\def\appendmask{\XKV@testopta{\XKV@testoptc\XKM@appendmask}}

% Query the mask for a key
\long\def\XKM@ifinmask[#1]#2#3{
  \XKV@checksanitizea{#1}\XKV@tempc
  \XKV@for@o\XKV@fams\XKV@tfam{
    \XKV@for@o\XKV@tempc\XKV@tempb{
      \xdef\XKM@hdr{\XKV@prefix\XKV@tfam @}
      \xdef\XKV@tempd{\XKV@prefix\XKV@tfam @\XKV@tempb}
      \@expandtwoargs\in@\XKV@tempd\XKM@mask
      \ifin@
        #2
      \else
        #3
      \fi
    }
  }
}
\def\ifinmask{\XKV@testopta{\XKV@testoptc\XKM@ifinmask}}

% A quick way to clear the mask
\def\clearmask{\global\let\XKM@mask\@empty}

% Now we override the definition of \XKV@setkeys
\long\def\XKM@setkeys[#1]#2{
  \XKV@checksanitizea{#2}\XKV@resb
  \let\XKV@resa\@empty
  \XKV@for@o\XKV@fams\XKV@tfam{
    \XKV@for@o\XKV@resb\XKV@tempb{
      \expandafter\XKV@g@tkeyname\XKV@tempb=\@nil\XKV@tempc
      % Remove spaces around key name
      \expandafter\KV@@sp@def\expandafter\XKV@tempc\expandafter{\XKV@tempc}
      \xdef\XKV@tempd{\XKV@prefix\XKV@tfam @\XKV@tempc}
      % If \XKV@tempd is in \XKM@mask,
      % remove the corresponding kv from \XKV@resb
      \@expandtwoargs\in@\XKV@tempd\XKM@mask
      \ifin@\else\XKV@addtolist@o\XKV@resa\XKV@tempb\fi
    }
  }
  \ifnum\XKV@depth=\z@\let\XKV@rm\@empty\fi
  \expandafter\XKV@s@tkeys\expandafter{\XKV@resa}{#1}
  \let\CurrentOption\@empty
}

% A dispatcher to \XKM@setkeys or \XKV@setkeys
\long\def\XKM@setkeys@dispatch{
  \xdef\XKV@tempa{\xkeymask@prefix @}
  \ifx\XKV@prefix\XKV@tempa
```

```
      \expandafter\XKM@setkeys
    \else
      \expandafter\XKV@setkeys
    \fi
  }

  % Finally, override \setkeys itself to call the dispatcher
  \def\setkeys{\XKV@testopta{\XKV@testoptc\XKM@setkeys@dispatch}}
  \makeatother
\fi
```