The **somedefs** toolkit package

Alan Jeffrey

long time ago in a different century...

This file is maintained by the \LaTeX Project team. Bug reports can be opened (category **tools**) at [https://latex-project.org/bugs.html](https://latex-project.org/bugs.html).

**Overview**

This is an example ‘programmers toolkit’ package, for use by package writers. It allows package writers to provide options which switch definitions on and off. For example, a package **fred** might define a large number of commands, including \texttt{\textbackslash foo} and \texttt{\textbackslash baz}, so:

\texttt{\usepackage{fred}}

would use a lot of memory, even if \texttt{\textbackslash foo} and \texttt{\textbackslash baz} were the only commands needed. However, if the author of **fred** used the **somedefs** package, then the user would be able to say:

\texttt{\usepackage[only,foo,baz]{fred}}

and only the commands \texttt{\textbackslash foo} and \texttt{\textbackslash baz} would be defined.

To use the **somedefs** package in your own packages or classes, you say:

\texttt{\RequirePackage{somedefs}}

You can then use four new commands:

- \texttt{\textbackslash UseAllDefinitions} which says that all the commands in the file should be defined.

- \texttt{\textbackslash UseSomeDefinitions} which says that only the commands specified by \texttt{\textbackslash UseDefinition} should be defined.

- \texttt{\textbackslash UseDefinition{\texttt{name}}} which says that the command \texttt{name} should be defined.

- \texttt{\textbackslash ProvidesDefinition{\texttt{definition}}} which provides one definition, of the form \texttt{\definingcommand{\command}}...
For example, the package Fred could say:

\RequirePackage{somedefs}
\UseAllDefinitions
\DeclareOption{only}{\UseSomeDefinitions}
\DeclareOption*{\UseDefinition{\CurrentOption}}
\ProcessOptions
\ProvidesDefinition{\newcommand{\foo}{...}}
\ProvidesDefinition{\newcommand{\baz}{...}}

One of the commands \UseAllDefinitions or \UseSomeDefinitions should always be used. You may have some commands which need other commands, in which case you have to declare the options by hand. For example, if the command \bar needs the command \foo, you could say:

\DeclareOption{bar}{\UseDefinition{bar}\UseDefinition{foo}}

For a longer example of the use of the somedefs package, look at the rawfonts package.

Implementation

The driver for the documentation you're now reading.

\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{somedefs}[1994/06/01 v0.03 Toolkit for optional definitions]

The package works by having \UseDefinition{\langle name\rangle} define \name to be \unprovides@definition. If \UseSomeDefinitions has been called, then \ProvidesDefinition looks to see if \name is \unprovides@definition. If \UseAllDefinitions has been called, then \ProvidesDefinition does nothing. If neither has been called, then \ProvidesDefinition produces an error message.

\def\UseSomeDefinitions{%
  \let\ProvidesDefinition@providesdefinition\unprovides@definition
}
\def\UseAllDefinitions{%
  \let\ProvidesDefinition@firstofone\unprovides@definition
}
\def\UseDefinition#1{%
  \expandafter\let\csname#1\endcsname\unprovides@definition
}
\def\ProvidesDefinition#1{%
  \PackageError{somedefs}{No \noexpand\UseSomeDefinitions or \string\UseAllDefinitions}{}
}
\begin{verbatim}
\def\@providesdefinition#1{%\@provides@definition#1\relax
\@provides@definition}
\def\@provides@definition#1#2#3\@provides@definition{%
  \ifx#2@unprovided@definition
    #1#2#3%n
  \fi
\def\@unprovided@definition{%
  \PackageError{somedefs}{Package 'somedefs' error: this command was never defined}{You have requested a command which does not exist.}%
}\@onlypreamble\UseSomeDefinitions
\@onlypreamble\UseAllDefinitions
\@onlypreamble\UseDefinition
\@onlypreamble\ProvidesDefinition
\@onlypreamble\@providesdefinition
\@onlypreamble\@provides@definition

That's it!
\end{verbatim}

\langle /package \rangle