The \texttt{l3docstrip} package
Code extraction and manipulation

The \LaTeX{}3 Project\textsuperscript{*}
Released 2020-05-05

\section{Extending DocStrip}

The \texttt{l3docstrip} module adds \LaTeX{}3 extensions to the DocStrip program for extracting code from \texttt{.dtx}. As such, this documentation should be read along with that for \texttt{DocStrip}.

\section{Internal functions and variables}

An important consideration for \LaTeX{}3 development is separating out public and internal functions. Functions and variables which are private to one module should not be used or modified by any other module. As \TeX{} does not have any formal namespaces system, this requires a convention for indicating which functions in a code-level module are public and which are private.

Using \texttt{l3docstrip} allows internal functions to be indicated using a “two part” system. Within the \texttt{.dtx} file, internal functions may be indicated using \texttt{@@} in place of the module name, for example

\begin{verbatim}
\cs_new_protected:Npn @@_some_function:nn #1#2 {
  \% Some code here
}
tl_new:N \l_@@_internal_tl
\end{verbatim}

To extract the code using \texttt{l3docstrip}, the “guard” concept used by \texttt{DocStrip} is extended by introduction of the syntax \texttt{%<@@=(module)}. The \texttt{(module)} name then replaces the \texttt{@@} when the code is extracted, so that

\begin{verbatim}
%<package>
%<@@=foo>
\cs_new_protected:Npn \@@_some_function:nn #1#2 {
  \% Some code here
}
dl_new:N \l_@@_internal_tl
%</package>
\end{verbatim}

\textsuperscript{*}E-mail: latex-team@latex-project.org
is extracted as

\cs_new_protected:Npn \__foo_some_function:nn #1#2
  {
    \% Some code here
  }
\tl_new:N \l__foo_internal_tl

where the __ indicates that the functions and variables are internal to the foo module.

Use @@@ to obtain @@ in the output (@@@@@ to get @@@@). For longer pieces of code
the replacement can be completely suppressed by giving an empty module name, namely
using the syntax \%<@@=>.