NAME

illit − Illiterate Programming File Format

DESCRIPTION

This manual page describes how to write source code that **illit** can use to generate literate program listings.

SOURCE FILES

An **illit** source file is a normal program source file with specially formatted comments. This has the great advantage that no preprocessing is necessary before the source file can be interpreted or compiled.

Since the comments expand to LaTeX code care must be taken using certain characters that LaTeX considers significant, in particular '\' and '&'. These two character may be escaped by preceding them with a backslash. See the LaTeX manual for other special characters. It is beyond the scope of this manual page to teach LaTeX.

illit recognizes three special types of comment: shorthand tags, sectioning tags, and metadata tags. Comments must be on a line by themselves, lines containing both code and comments are not currently supported.

SHORTHAND

Shorthand tags consist of a word in capitals optionally followed by a colon and a string of text up until the end of the line. These get expanded into:

\todo{*tag***}{***description***}**

The template must supply a suitable definition for **\todo**. The following shorthand tags are supported (along with suggested uses):

TODO Something to fix in the future.

XXX Something's wrong/missing here.

FIXME Fix me now!

NOTE Special note about this piece of code.

STANDARD METADATA TAGS

All tags apart from shorthand tags consist of two lower-case letters following a full stop. They must always appear at the beginning of a comment (leading whitespace is permitted).

Metadata tags append the rest of the line following the tag to the document metadata item with the same name as them. The template may include metadata as it sees fit. Any two-letter tag name may be used apart from the sectioning and special tag names listed below but the following are standard and may be expected to be understood by most templates:

.tt The title of the file (not the file name).

.au The author(s) of the file. Separate multiple authors with "**\and**".

.ve The version number of the file.

.ch This tag is reserved for recording the change history of a file. Its format is not currently defined.

SPECIAL TAGS

Two tags are treated differently. "**.da**" takes an ISO date (YYYY/MM/DD) as its argument and sets three separate items of metadata: **dyr** (year), **dmo** (month), and **dda** (day of month). "**.fi**" is set automatically to the name of the input file (or blank if the input is stdin) unless it is set using the **--meta** flag on the com-

mand line (it can also be set in a `.illitrc` file, but this is generally not useful) but can be overridden in a source file. When set automatically any special LaTeX characters (backslashes, underscores, etc.) will be escaped.

Note that neither of the special tags append their argument to any previously set value of the tag, they always overwrite it.

SECTIONING TAGS

Sectioning tags are used to break the source file up into LaTeX sections so that a table of contents can be produced. Any trailing full stop on a sectioning tag will be removed.

.sa, **.sb**, **.sc**, **.sd**, **.se**
: Section headers. These get translated into LaTeX macros from "**\seca{}**" to "**\sece{}**" with the argument of the macro set to the contents of the line after the sectioning tag. The template should define these macros in terms of chapters and sections as appropriate. Use **\seca** as the highest level heading and **\sece** as the lowest.

.fn *name / description*
: Use this tag to start a function definition. The function name must be a single word with no whitespace while the description may contain white space and is terminated by the end of the line. This tag is translated into "**\func{***name***}{***description***}**".

CROSSED-OUT LINES

To comment out a line of source code, start the comment with 'X' followed by a space. These lines will then be displayed as crossed-out.

LITERALS IN COMMENTS

Templates must load the **shortvrb** package. This is configured by **illit** so that the vertical bar character may be used to surround literal text (which will then be displayed in a typewriter font) in comments.

TEMPLATES

Templates are standard LaTeX files which may contain two special tags. Each of these tags may appear only once in the template.

.me *tagname*, *...*
: Generates a "**\meta***tagname*" macro for each tag name listed containing the contents of that tag. If a tag is not set by the source file then the macro will be empty. Use a package such as **ifthen** to detect blank tags and supply defaults if required.

.bd
: Include the processed source file. If the template does not contain this tag then the processed source will be appended to the end of the template.

TEMPLATE REQUIREMENTS

illit assumes that certain LaTeX packages are available and that the template defines certain macros. See the default template included in the **illit** distribution for example definitions.

Packages

The **listings** package must be loaded and configered for the display of source code. If required it may be set to pretty print code. The **shortvrb** package must be loaded so that literals can be written in comments. Do not configure this package since **illit** does this itself.

Environments

Comments are indented to match the source code using an environment called **indented**. This must take a single argument of the number of tab stops that the comment should be indented by. If non-indented com-

ments are required then define this as a dummy environment.

Macros

The template must define the following macros:

\crossedout{*text***}**
: Display the argument crossed-out.

\seca{*heading***} ... \sece{***heading***}**
: Section headers.

\func{*name***}{***description***}**
: Function header.

\todo{*tag***}{***description***}**
: Shorthand.

SEE ALSO

`illit`(1)

AUTHOR

Asher Hoskins