

## NAME

typog-grep – specialized grep for typog-inspect elements in LaTeX log files

## SYNOPSIS

**typog-grep** -a|--all|--any [*OPTION*...] *LOG-FILE*...

**typog-grep** [*OPTION*...] *REGEXP LOG-FILE*...

The first form, “discovery mode”, shows all *IDs* of

```
<typog-inspect id="ID" ...>
```

elements in *LOG-FILE*.

The second form shows the contents, *LOG-DATA*, of the elements

```
<typog-inspect id="ID" ...>
```

```
LOG-DATA
```

```
</typog-inspect>
```

whose *IDs* match *REGEXP* in *LOG-FILE*.

If no *LOG-FILE* is given read from *stdin*. The filename – is synonymous to *stdin*.

## DESCRIPTION

**typog-grep** is a tailored post-processor for LaTeX *log* files and the `typoginspect` environment as provided by the LaTeX package `typog`. It shares more with the venerable **sgrep** <<https://www.cs.helsinki.fi/u/jjaakkol/sgrep.html>> than with POSIX **grep** <<https://pubs.opengroup.org/onlinepubs/9699919799/utilities/grep.html>>.

In the LaTeX source file the user brackets her text or code in a `typoginspect` environment:

```
\begin{typoginspect}{ID}
```

```
TEXT-OR-CODE-TO-INVESTIGATE
```

```
\end{typoginspect}
```

where *ID* is used to identify one or more bracketed snippets. *ID* does not have to be unique. The *REGEXP* mechanism makes it easy to select groups of related *IDs* if they are named accordingly.

In *LOG-FILE* the result of the environment shows up, packed with tracing information, as

```
<typog-inspect id="ID" job="JOB-NAME" line="LINE-NUMBER" page="PAGE-NUMBER">
```

```
LOG-DATA
```

```
</typog-inspect>
```

where all the capital-letter sequences are meta-variables and in particular *JOB-NAME* is the expansion of `\jobname`, *LINE-NUMBER* is the LaTeX source file line number of the beginning of the `typoginspect` environment, and *PAGE-NUMBER* is the page where the output of `TEXT-OR-CODE-TO-INVESTIGATE` occurs.

**typog-grep** reveals the contents of *LOG-FILE* between `<typog-inspect id="ID" ...>` and `</typog-inspect>` excluding the XML-tags themselves. Access the *JOB-NAME*, *LINE-NUMBER*, and *PAGE-NUMBER* with the commandline options **--job-name**, **--line-number**, and **--page-number**, respectively. Use **--id** to show the name of the *IDs* that matched *REGEXP*.

`typoginspect` environments can be nested. **typog-grep** respects the nesting, i.e., if the *ID* of the nested environment does not match *REGEXP* it will not be included in the program’s output.

## OPTIONS

The list of options is sorted by the names of the long options.

**-a, --all, --any**

ID-discovery mode: Discover all `typog-inspect` elements independent of any matching patterns and print their *IDs*. The results are printed in their order of occurrence in the respective *LOG-FILES*. Pipe the output into **sort** to get alphabetically ordered *IDs*.

Augment with options **--job-name**, **--line-number**, **--log-line-number**, or **--page-number** for more information.

**--color, colour WHEN**

Colorize specific log contents for the matching *IDs*. The argument *WHEN* determines when to apply color: `always`, `never`, or `auto`. The setting `auto` checks whether standard output has been redirected. This is the default.

**-C, --config KEY=VALUE[:KEY=VALUE[...]]**

Set one or more configuration *KEY* to *VALUE* pairs. See section "CONFIGURATION" for a description of all available configuration items. Use option **--show-config** to display the default configuration.

**--debug**

Turn on debug output on *stderr*.

**-E, --encoding ENCODING**

Set the *ENCODING* of *LOG-FILE* for the translation to UTF-8. The default is unset.

Use this option to get rid of pesky "<HEX-DIGITS>" escapes on UTF-8 terminals. See option **--show-encodings** for the known encodings and `Encode::Supported` for a summary of all encodings. See also section "Some Common Encodings".

Apply **iconv** <<https://pubs.opengroup.org/onlinepubs/9699919799/utilities/iconv.html>> (POSIX) or **recode** <<https://github.com/rrthomas/recode/>> (GNU) on *LOG-FILE* before this tool to avoid having to use option **--encoding**.

**-h, --help**

Display brief help then exit.

**-i, --[no-]id**

Print the actual ID-name that matched *REGEXP*. Control the appearance of the matching *ID* with configuration item `id-heading`.

**-y, --[no-]ignore-case**

Match *IDs* while ignoring case distinctions in patterns and data.

**-j, --[no-]job-name**

Print the `\jobname` that **latex** associated with the input file.

**-n, --[no-]line-number**

Print the line number where the `typoginspect` environment was encountered in the LaTeX source file.

**-N, --[no-]log-line-number**

Print the line number of the *log*-file where the current line was encountered.

**-p, --[no-]page-number**

Print page number where the contents of the `typoginspect` environment starts in the typeset document.

- P, --[no-]pager**  
Redirect output from *stdout* to the configured pager.
- show-config**  
Show the default configuration and exit.
- show-encodings**  
Show all known encodings and exit.
- V, --version**  
Show version information and exit.
- w, --[no-]word-regexp**  
Match only whole words.

## CONFIGURATION

- id-format=FORMAT**  
Control the *FORMAT* for printing matching ids in inline-mode, where *FORMAT* is passed to Perl's `printf`. Default: `%s:`.
- id-heading=0|1**  
Choose between printing the matching *IDs* with option **--id**: Inline (0) or heading before the matching data (1). Default: 0.
- id-heading-format=FORMAT**  
Control the *FORMAT* for printing matching *IDs* in heading-mode, where *FORMAT* is passed to Perl's `printf`. Default: `--> %s <--`.
- id-indent=INDENT**  
Indentation of nested typog-inspect tags. Only used in “discovery mode” (first form), i.e., if **--all** is active. Default: 8.
- id-max-length=MAXIMUM-LENGTH**  
Set the maximum length of a matching *ID* for printing. If a matching *ID* exceeds this length it will be truncated and the last three characters (short of *MAXIMUM-LENGTH*) will be replaced by dots. Default: 40.
- line-number-format=FORMAT**  
Control the *FORMAT* for printing TeX source line numbers, where *FORMAT* is passed to Perl's `printf`. Default: `%5d`.
- log-line-number-format=FORMAT**  
Control the *FORMAT* for printing log line numbers, where *FORMAT* is passed to Perl's `printf`. Default: `%6d`.
- page-number-format=FORMAT**  
Control the *FORMAT* for printing page numbers, where *FORMAT* is passed to Perl's `printf`. Default: `[%3d]`.
- pager=PAGER**  
Name of pager application to pipe output into if run with option **--pager**. Default: `less`.
- pager-flags=FLAGS**  
Pass *FLAGS* to *PAGER*. Default: `--quit-if-one-screen`.

## Color Configuration

For the syntax of the color specifications consult the manual page of `Term::ANSIColor(pm)`.

`file-header-color`

Color of the filename header.

`fill-state-color`

Color of the messages that report “Underfull hbox” or “Overfull hbox”.

`first-vbox-color`

Color of the first vbox on a page.

`font-spec-color`

Color of font specifications.

`horizontal-break-candidate-color`

Color of lines with horizontal-breakpoint candidates @.

`horizontal-breakpoint-color`

Color of lines with horizontal breakpoints @@.

`id-color`

Color of matching *IDs* when printed inline.

`id-heading-color`

Color of matching *IDs* when printed in heading form.

`line-break-pass-color`

Color of the lines showing which pass (e.g., @firstpass) of the line-breaking algorithm is active.

`line-number-color`

Color of TeX-source-file line numbers.

`log-line-number-color`

Color of log-file line numbers.

`math-color`

Color used for math expressions including their font specs.

`page-number-color`

Color of page numbers of the final output.

`tightness-color`

Color of lines with Tight/Loose hbox reports.

`vertical-breakpoint-color`

Color of possible vertical breakpoints.

## Brief summary of colors and attributes

### Foreground Color

black, red, green, yellow, blue, magenta, cyan, white,

Prefix with `bright_` for high-intensity or bold foreground.

### Foreground Grey

grey0, ..., grey23

### Background Color

`on_black`, `on_red`, `on_green`, `on_yellow`, `on_blue`, `on_magenta`, `on_cyan`,  
`on_white`

Replace `on_` with `on_bright_` for high-intensity or bold background.

### Background Grey

`on_grey0`, ..., `on_grey23`

### Text Attribute

`bold`, `dark`, `italic`, `underline`, `reverse`

## Some Common Encodings

The following list shows some encodings that are suitable for option `--encoding`.

### Latin-1, Western European

`iso-8859-1`, `cp850`, `cp860`, `cp1252`

### Latin-2, Central European

`iso-8859-2`, `cp852`, `cp1250`

### Latin-3, South European (Esperanto, Maltese)

`iso-8859-3`

### Latin-4, North European (Baltics)

`iso-8859-4`

### Cyrillics

`iso-8859-5`, `cp855`, `cp866` (Ukrainian), `cp1251`

### Arabic

`iso-8859-6`, `cp864`, `cp1006` (Farsi), `cp1256`

### Greek

`iso-8859-7`, `cp737`, `cp1253`

### Hebrew

`iso-8859-8`, `cp862`, `cp1255`

### Turkish

`iso-8859-9`, `cp857`, `cp1254`

### Nordic

`iso-8859-10`, `cp865`, `cp861` (Icelandic)

### Thai

`iso-8859-11`, `cp874`

### Baltic

`iso-8859-13`, `cp775`, `cp1257`

### Celtic

`iso-8859-14`

### Latin-9 (sometimes called Latin0)

`iso-8859-15`

```
Latin-10
iso-8859-16
```

## EXIT STATUS

The exit status is 0 if at least one *ID* matched *REGEXP*, 1 if no *ID* matched *REGEXP*, and 2 if an error occurred.

## CAVEATS

The end tag `</typog-inspect>` sometimes gets placed too early in the output and the trace *seems* truncated. However, LaTeX reliably logs the requested the trace information, but the write operations for trace data and the code which is used to print the end tag are not synchronized.

## SEE ALSO

**grep(1)**, **printf(3)**, **Encode::Supported(pm)**, **Term::ANSIColor(pm)**