

Edi: The Enlightenment IDE



Code Editor to Productivity Tool

Andrew Williams



Overview

- An IDE written using EFL for EFL
- Supporting best practice in Software Engineering
- Our largest Elementary app
- A lot of hard work!

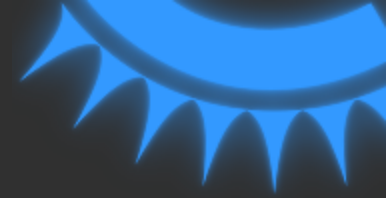
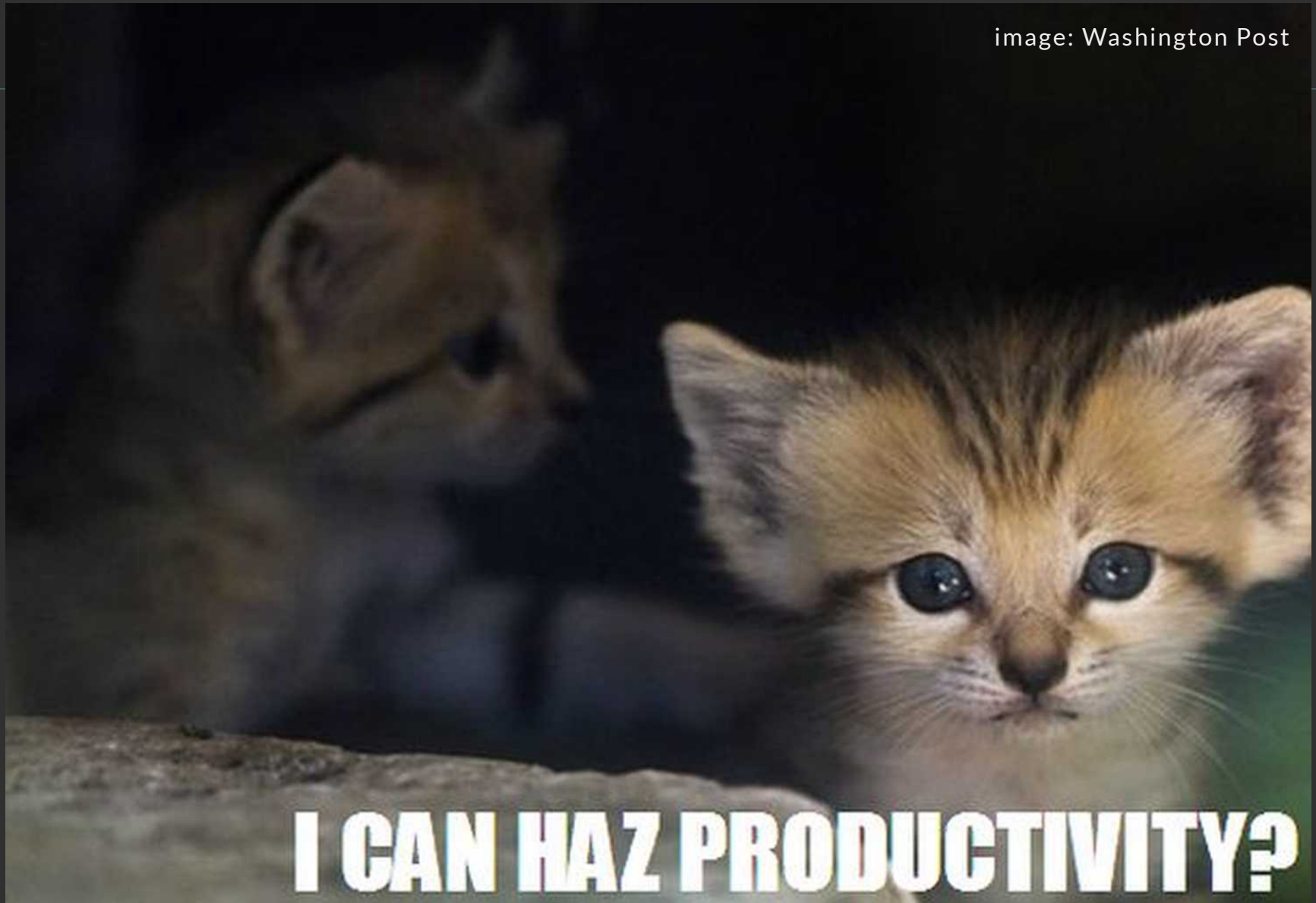


image: Washington Post



I CAN HAZ PRODUCTIVITY?

Busy Year 😄



Releases (1)

0.5

- Autosuggest
- Inline docs
- Smart Indent
- Python support
- TODO/FIXME panel

0.5.1

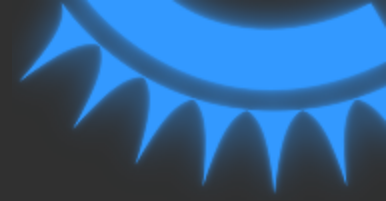
- Bug fixes from latest EFL - a long wait until 0.6



Releases (2)

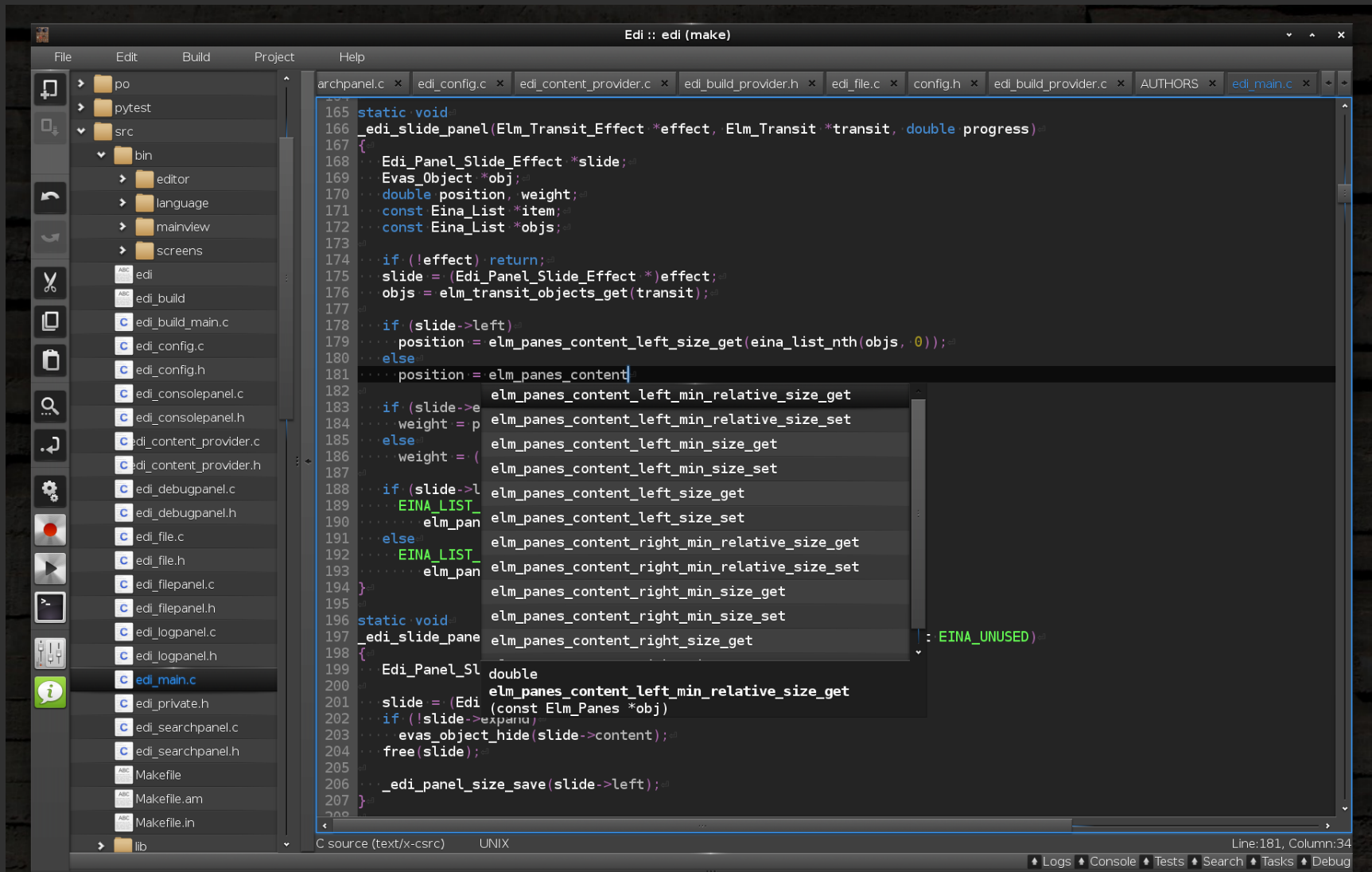
0.6-pre

- Debugger
- Source Control
- Snippets
- Project skeletons

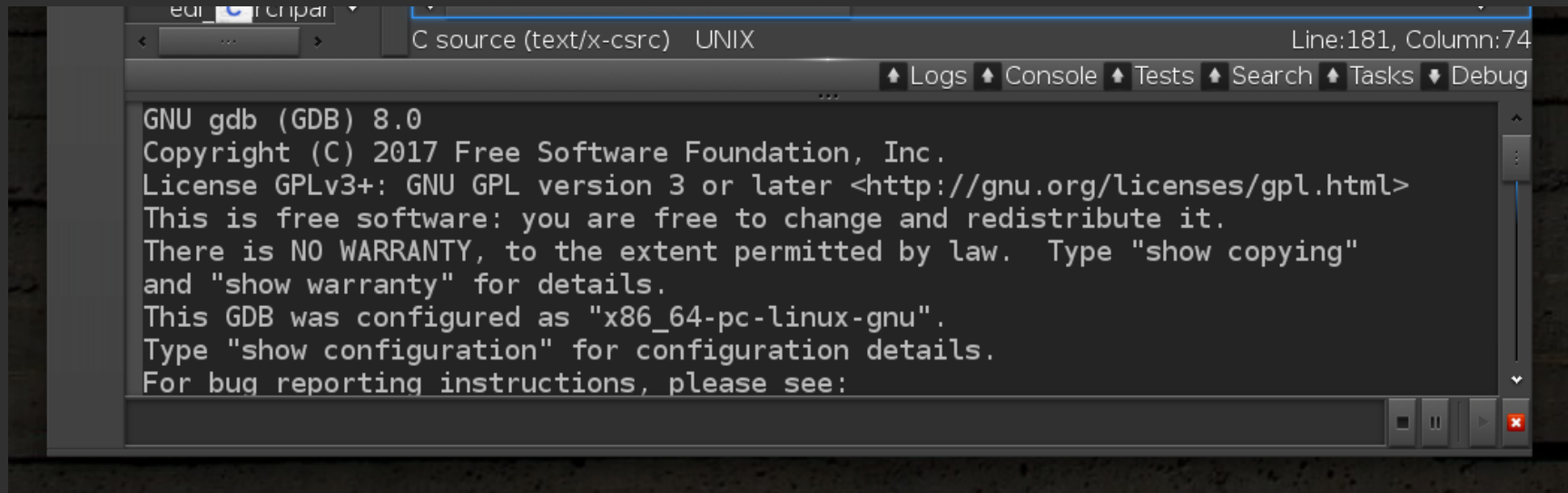


Feature Highlights

Auto-suggest



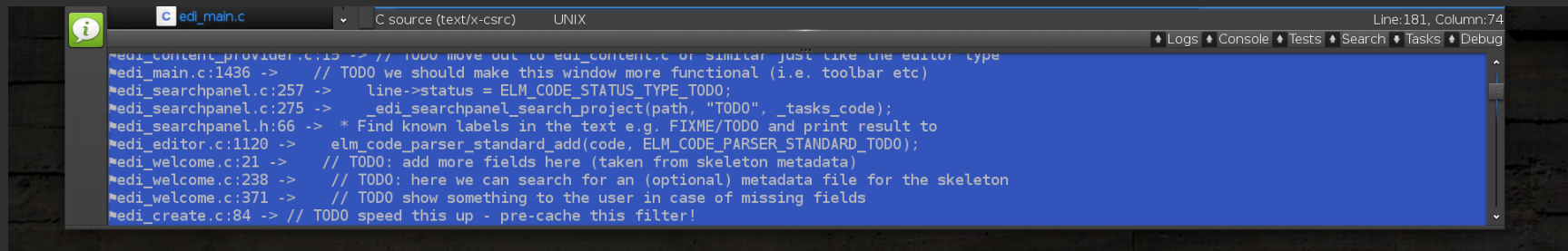
Debug



The screenshot shows a debugger window with a dark theme. The title bar indicates the file is 'C source (text/x-csrc)' on a 'UNIX' system, with the cursor at 'Line:181, Column:74'. The window has tabs for 'Logs', 'Console', 'Tests', 'Search', 'Tasks', and 'Debug'. The 'Console' tab is active, displaying the GNU gdb (GDB) 8.0 splash screen. The text in the console reads: 'GNU gdb (GDB) 8.0', 'Copyright (C) 2017 Free Software Foundation, Inc.', 'License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>', 'This is free software: you are free to change and redistribute it.', 'There is NO WARRANTY, to the extent permitted by law. Type "show copying" and "show warranty" for details.', 'This GDB was configured as "x86_64-pc-linux-gnu".', 'Type "show configuration" for configuration details.', and 'For bug reporting instructions, please see:'. At the bottom right of the console, there are icons for a window, a pause, a play, and a close button.

```
GNU gdb (GDB) 8.0
Copyright (C) 2017 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-pc-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
```

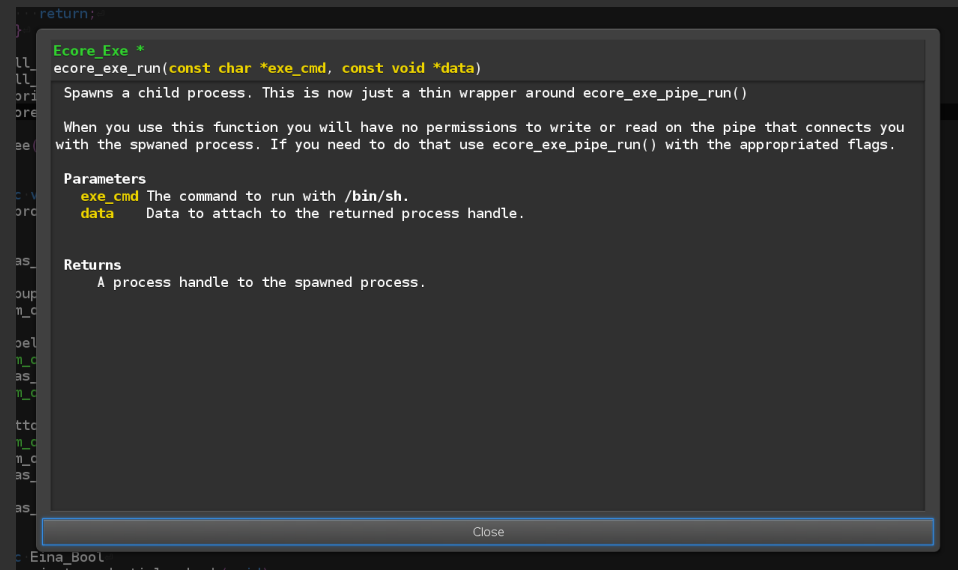
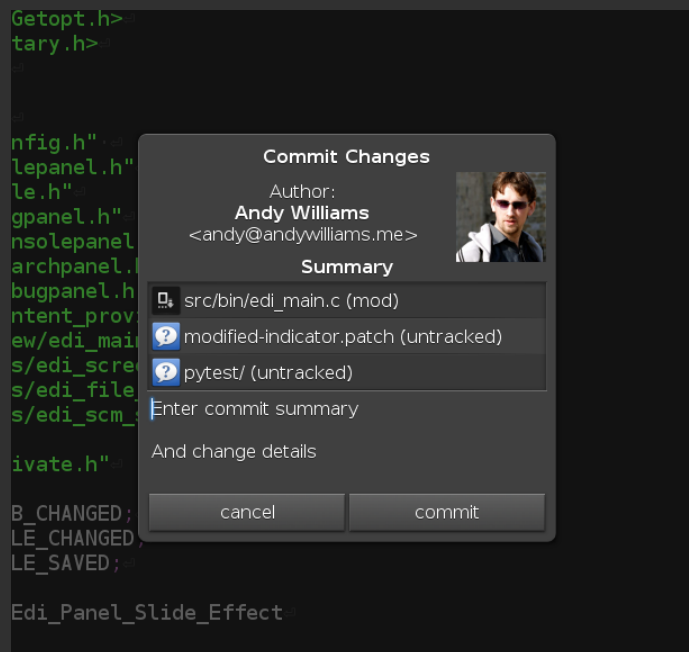
TODO / FIXME



The screenshot shows a code editor window with a dark theme. The title bar indicates the file is 'C source (text/x-csrc)' on a 'UNIX' system, with the cursor at 'Line:181, Column:74'. The window has tabs for 'Logs', 'Console', 'Tests', 'Search', 'Tasks', and 'Debug'. The 'Console' tab is active, displaying a list of TODO and FIXME comments. The comments are as follows: 'edi_content_provider.c:15 -> // TODO move out to edi_content.c or similar just like the editor type', 'edi_main.c:1436 -> // TODO we should make this window more functional (i.e. toolbar etc)', 'edi_searchpanel.c:257 -> line->status = ELM_CODE_STATUS_TYPE_TODO;', 'edi_searchpanel.c:275 -> _edi_searchpanel_search_project(path, "TODO", _tasks_code);', 'edi_searchpanel.h:66 -> * Find known labels in the text e.g. FIXME/TODO and print result to', 'edi_editor.c:1120 -> elm_code_parser_standard_add(code, ELM_CODE_PARSER_STANDARD_TODO);', 'edi_welcome.c:21 -> // TODO: add more fields here (taken from skeleton metadata)', 'edi_welcome.c:238 -> // TODO: here we can search for an (optional) metadata file for the skeleton', 'edi_welcome.c:371 -> // TODO show something to the user in case of missing fields', and 'edi_create.c:84 -> // TODO speed this up - pre-cache this filter!'. The code is highlighted in blue.

```
edi_content_provider.c:15 -> // TODO move out to edi_content.c or similar just like the editor type
edi_main.c:1436 -> // TODO we should make this window more functional (i.e. toolbar etc)
edi_searchpanel.c:257 -> line->status = ELM_CODE_STATUS_TYPE_TODO;
edi_searchpanel.c:275 -> _edi_searchpanel_search_project(path, "TODO", _tasks_code);
edi_searchpanel.h:66 -> * Find known labels in the text e.g. FIXME/TODO and print result to
edi_editor.c:1120 -> elm_code_parser_standard_add(code, ELM_CODE_PARSER_STANDARD_TODO);
edi_welcome.c:21 -> // TODO: add more fields here (taken from skeleton metadata)
edi_welcome.c:238 -> // TODO: here we can search for an (optional) metadata file for the skeleton
edi_welcome.c:371 -> // TODO show something to the user in case of missing fields
edi_create.c:84 -> // TODO speed this up - pre-cache this filter!
```

Screens



Contributors

16 total contributors!

Huge contributions from

- netstar (Al Poole)
- YOhoho (YeongJong Lee)

Edi ❤️s you champions!





The Crystal Ball



Roadmap

0.6 & 0.7

- Code folding
- Split pane
- Code formatting

1.0

- Refactoring
- Packaging

How to match EFL / Eo roadmap?



Encouraging new developers

More content for getting people started

Championing Software Engineering best practices

Community backing

- Include tooling in documentation
- Get to stable, modern API
- A "default" EFL dev setup



The importance of being Edi

Examples

The first project new developers meet *
Demonstrate the full power of EFL

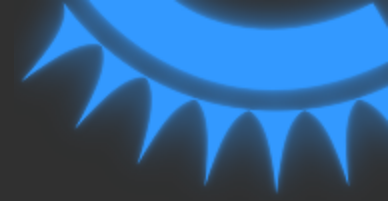
Features

Provide what people expect of an IDE
Break free of "EFL is difficult"

Time

I quit (my day job) 😎

Questions?



- @ajwillia_ms
- andy@ajwillia.ms
- <http://ajwillia.ms>

