CMDR - interactive command line interfaces in Rust

Mendelt Siebenga

CMDR

or Procedural macro's for fun and profit

About me

- Recovering Python programmer
- I programs Scala for money
- I play with Rust for fun

This presentation

- Part 1: Show off CMDR
- Part 2: How does it work
    - Procedural macros

Some Python (Cmd)

```python
import cmd

class HelloWorld(cmd.Cmd):
    """Simple command processor example."""

    def do_greet(self, line):
        print "hello"

    def do_EOF(self, line):
        return True

if __name__ == '__main__':
    HelloWorld().cmdloop()
```

Rust ‘problems’

- No class based inheritance
  - But we do have traits
- No introspection/reflection
  - But we do have macros

Demo time

- Simple commands
- Help function
- Changing the prompt
- Command scopes
- Hooks

Planned functionality for 1.0

- More examples and documentation
- Tab completion
    - Commands
    - Parameters
- Typed argument parsing

Some usage suggestions

- Old school text based adventure/dungeon crawlers
- Management application for your system
- Test harness for an API

More information

- crates.io crates.io/crates/cmdr
- code github.com/mendelt/cmdr
    - with examples
- docs.rs/cmdr
- cargo generate template: github.com/mendelt/cmdr-template
- This presentation: github.com/mendelt/cmdr-presentation

Macro's

- macro_rules! macros
  - println!
  - vec[]
- derive macros
  - serde
- procedural macros

Procedural macro code example

- Cmdr by hand
- cargo expand

Questions?