Tools That Spark Joy: Lessons Learned From the Rust Ecosystem That Can Be Adopted Elsewhere

Lotte Steenbrink
TOOLS THAT SPARK JOY

lessons learned from the Rust ecosystem that can be adopted elsewhere

LOTTE STEENBRINK, FERROUS SYSTEMS
Hello!

LOITTE STEENBRINK, SOFTWARE ENGINEER

- IoT, constrained networks, embedded systems
- IETF
- C & C++

AT FERROUS SYSTEMS
• Build Trust
• Reduce Friction and Fear
• Spark Joy

OUTLINE

1. built-in Rust tools
2. 3rd party open source tooling
3. lessons learned
Rust is a systems programming language that is optimised for speed, memory safety and fearless concurrency.
Rust is a systems programming language that is optimised for speed, memory safety and fearless concurrency.

RUST IS AN EMPATHIC SYSTEMS PROGRAMMING LANGUAGE THAT IS DETERMINED TO NOT LET ME SHOOT MYSELF IN THE FOOT.
let me show you what I mean.

```rust
fn main() {
    let answer = 0;
    // do some more calculations ...
    answer = 42;
    println!("The Answer to the Ultimate Question of Life, the Universe, and Everything is: {}", answer);
}
```
let me show you what I mean.

```rust
fn main() {
    let answer = 0;
    // do some more calculations ...
    answer = 42;
    println!("The Answer to the Ultimate Question of Life, the Universe, and Everything is: {}, answer");
}
```

**error[E0384]:** cannot assign twice to immutable variable `answer`

```
--- src/bin/deep_thought.rs:10:5
8 | let answer = 0;
  |     ------
  |     first assignment to `answer`
  |     help: make this binding mutable: `mut answer`
9 | // do some more calculations ...
10 | answer = 42;
   | ^^^^^^^^^^^ cannot assign twice to immutable variable
```

**error:** aborting due to previous error

For more information about this error, try `rustc --explain E0384`.

[try it out on play.rust-lang.org](http://play.rust-lang.org)
error[E0384]: cannot assign twice to immutable variable `answer`

```rust
let answer = 0;
first assignment to `answer`
help: make this binding mutable: `mut answer`
```

error: aborting due to previous error

For more information about this error, try `rustc --explain E0384`.

error[E0384]: cannot assign twice to immutable variable `answer`
  --> src/bin/deep_thought.rs:10:5
   | 8 | let answer = 0;
   | | -----
   | | |
   | | first assignment to `answer`
   | | help: make this binding mutable: `mut answer`
 9 | // do some more calculations ...
10 | answer = 42;
   | ^^^^^^^^^^^ cannot assign twice to immutable variable

error: aborting due to previous error

For more information about this error, try `rustc --explain E0384`.
error[E0384]: cannot assign twice to immutable variable `answer`
--> src/bin/deep_thought.rs:10:5
|  
| 8 | let answer = 0;
| | ------
| | |
| | first assignment to `answer`
| | help: make this binding mutable: `mut answer`
9 | // do some more calculations ...
10 | answer = 42;
| ^^^^^^^^^^ cannot assign twice to immutable variable

error: aborting due to previous error

For more information about this error, try `rustc --explain E0384`.
error[E0384]: cannot assign twice to immutable variable `answer`

-- src/bin/deep_thought.rs:10:5
| let answer = 0;
| | ------
| | |
| | first assignment to `answer`
| | help: make this binding mutable: `mut answer`
| // do some more calculations ...
| answer = 42;
| ^^^^^^^^^^ cannot assign twice to immutable variable

error: aborting due to previous error

For more information about this error, try `rustc --explain E0384`.
error[E0384]: cannot assign twice to immutable variable `answer`
  --> src/bin/deep_thought.rs:10:5
   | let answer = 0;
   |    ------
   |    |
   |    first assignment to `answer`
   |    help: make this binding mutable: `mut answer`
  9 | // do some more calculations ...
  10 | answer = 42;
      ^^^^^^^^^^^ cannot assign twice to immutable variable

error: aborting due to previous error

For more information about this error, try `rustc --explain E0384`.

---

By whitespace:

```rust
let answer = 0;
```

First assignment to `answer`.

```rust
answer = 42;
```

Cannot assign twice to immutable variable `answer`.
error[E0384]: cannot assign twice to immutable variable `answer`
---> src/bin/deep_thought.rs:10:5
|
8 | let answer = 0;
|-------
| |
| first assignment to `answer`
| help: make this binding mutable: `mut answer`
9 | // do some more calculations ...
10 | answer = 42;
| ^^^^^^^^^^^ cannot assign twice to immutable variable

error: aborting due to previous error

For more information about this error, try `rustc --explain E0384`.
Error[E0384]: cannot assign twice to immutable variable `answer`

```rust
let answer = 0;
```

#### Help:

- `mut answer`

```rust
answer = 42;
```

For more information about this error, try `rustc --explain E0384`. 

https://blog.rust-lang.org/2016/08/10/Shape-of-errors-to-come.html
ready, set, rust
ready, set, rust

- ferrous cargo new --lib xoxo
  Created library `xoxo` package
- ferrous cd xoxo
- xoxo git:(master) x tree

  - Cargo.toml
  - src
    - lib.rs

1 directory, 2 files
- xoxo git:(master) x
when the grass is greener on the other side, accept their gardening tips
not invented here

```rust
[target.\'cfg(all(target_arch = "arm", target_os = "none"))\']
runner = "probe-run --chip nRF52840_xxAA"
```

```bash
~ cargo install probe-run
```

Cargo.toml

```toml
[profile.dev]
debug = 1
```

./cargo.config

```toml
profile.dev

debug = 1
```
Make the right thing to do easy

– Ashley Williams
make the right thing to do enticing
make the right thing to do
enticing

```rust
#0
#1, #2, 30
#3, 01000111
```

```
defmt::info!("Hello 👋😊");
defmt::warn!("{=?}",
    Person { name: interned_name,
        age: 30 } );
defmt::error!("{0=0..3:x}",
    0b0100_0111u8);
```

```
INFO Hello 👋😊
WARN { name: Lotte, age: 30 }
ERROR 0x7
```
LESSONS LEARNED, THE LISTICLE™
PRESENTATION IS KEY

- delete noise
- whitespace groups information
- carefully emphasize
PUT HELP WHERE YOUR AUDIENCE NEEDS IT

- reduce context switches
- show and tell
- ❤️ copy-paste-able solutions
IF YOU HAD TO SUFFER THROUGH IT, MAKE SURE NOBODY ELSE HAS TO

- Explain it to the human, not their computer
- Automate setup steps
LEAD BY EXAMPLE

• set a standard to live up to and copy from
• *if possible*, consider 🌟joy🌟 at the design stage
• sponsor open source tools that make your life easier
STEAL GOOD IDEAS
KEEP THE HAPPY PATH CONSISTENT
MAKE THE RIGHT THING TO DO EASY

– Ashley Williams
MAKE THE RIGHT THING TO DO ENTERICING
DEAR INTERVIEWEES: THANK YOU

Florian Gilcher, Jorge Aparicio, James Munns,
Jan-Erik Rediger
**Sources**

🔗 click on each link to get to the source!

### TOOLS

**knurling-rs** (sponsor us!)

**defmt** | postform, a defmt-inspired C++ library

**cargo-watch** | entr

▶ Run button and *much* more IDE magic by **rust-analyzer**

**Rust Playground** | **C Playground**

### TALKS

**Bending the Curve: A Personal Tutor at Your Fingertips**

**The Anatomy of Error Messages in Rust**

**C++ & Rust: Access all Arenas**

**Zen and the Art of Convincing Your Company to Use Rust**

**RustConf 2020 Keynote**

**A Tale of Two Asyncs: Open Source Language Design in Rust and Node.js**
Sources

click on each link to get to the source!

TEXT

elm-lang.org/news/compiler-errors-for-humans
blog.rust-lang.org/2016/08/10/Shape-of-errors-to-come.html
usability improvements in gcc 9
gcc User Experience Guidelines
rustc diagnostics style guide
extending cargo with custom commands – 1 paragraph
Testing a Hardware Abstraction Layer (HAL) using defmt

OTHER

3D illustrations by craftwork.design
Chubby hands by pitch.com
TOOLS THAT SPARK JOY

lessons learned from the Rust ecosystem that can be adopted elsewhere

LOTTE STEENBRINK, FERROUS SYSTEMS
[dependencies]
addr2line = "0.14.1" ✔
ansi_term = "0.12.1" ✔
anyhow = "1.0.32" ✔
arrayref = "0.3.6" ✔
colored = "2.0.0" ✔
defmt-decoder = {
git = "https://github.com/knurling-rs/defmt",
tag = "defmt-decoder-v0.2.0",
difference = "2.0.0" ✔
gimli = "0.23.0" ✔
log = "0.4.11" ✔
}
# an addr2line trait is implement for a type in this particular version
object = "0.22.0" ❌ 0.23.0
probe-rs = "0.10.0" ✔
probe-rs-rtt = "0.10.0" ✔
rustc-demangle = "0.1.16" ✔
signal-hook = "0.3.4" ✔
structopt = "0.3.15" ✔
hidapi = "1.2.5" ✔
Esteban K - 🌸宯@home @ekuber - Dec 6, 2020

Just out of curiosity, was the error not clear for some reason? Could it be made easier to understand?
play.rust-lang.org/?version=stable...

error: expected `;`, found `}`
   --> src/main.rs:5:6
   5 | }
   | ^ help: add `;` here
   6 | }
   | - unexpected token

Jakub T. Jankiewicz (Ł) @j cubic

Replying to @ekuber and @IljiaMarijanov8

I'm not used to good error messages, so I didn't look at it actually LOL

8:54 AM · Dec 7, 2020 · Twitter Web App

2 Likes
beware of survivorship bias

"No need to address that, no-one is complaining about it."

1:19 AM - Feb 25, 2021 - Twitter for Android

21 Retweets  1 Quote Tweet  134 Likes