



Products

Why does Rust Closure take ownership before being called

[Login](#) [Sign up](#)

[Ask Question](#)

Asked 1 year, 7 months ago

Active 1 year, 6 months ago

Viewed 166 times



3



I'm going through the Rust book, and I'm on the chapter describing closures. I'm a bit confused about why the following example errors due to the closure taking ownership:

```
fn main() {  
    let x = vec![1, 2, 3];  
  
    let equal_to_x = move |z| z == x;  
  
    println!("{}", "can't use x here: {:?}", x);  
  
    let y = vec![1, 2, 3];  
  
    assert!(equal_to_x(y));  
}
```

Namely, why does `equal_to_x` take ownership of `x` before it's even called? Shouldn't the compiler know that the closure hasn't been called (because it is owned by `main`) and thus `x` can still be owned by the outside scope of `main`?

[rust](#) [closures](#)

Share

[Improve this question](#)

Follow

asked Jun 1 '20 at 3:06



[Kerastan](#)

33 • 3

2

why not ? it's more simple, keep the workflow clear. If you want change use `x` before just create your closure after.

– [Stargateur](#)

Jun 1 '20 at 3:23

If you're asking why its not moved lazily, that's not a thing that exists because lifetimes are checked statically. If you're asking why the compiler doesn't reorder your statements so that it works, that's also not a thing that Rust does.

– [kmdreko](#)

Jun 1 '20 at 3:36

For me it makes more sense to take the ownership before hand in closure so that if you need to mutate the state of variable in closure itself

– [Kartikeya Shama](#)

Jun 1 '20 at 5:13

You can pass the closure around before calling it. The closure may even outlive `x`. It's impossible to lazily take ownership once the closure is called for the first time.

– [Sven Mamach](#)

Jun 1 '20 at 8:15

[Add a comment](#)

1 Answer

[Active](#) [Oldest](#) [Votes](#)



5



Because a closure is essentially a struct with a function associated with it. I find it easier to think of closures like this:

```
struct MyClosure {  
    x: Vec<i32>  
};
```

```
impl MyClosure {  
    fn new(x: Vec<i32>) -> MyClosure {  
        z == self.x  
    }  
}
```

Your privacy

By clicking "Accept all cookies", you agree to the storing of cookies on your device to enhance site navigation, analyze site usage, and assist in our marketing efforts. (See our [Privacy Policy](#) for more details.)

Accept all cookies

Customize settings

If thought of like this, your code is equivalent to

```
let equal_to_x = MyClosure {  
  x  
};
```

so you can see why the move has happened.

[Share](#)

[Improve this answer](#)

[Follow](#)

[edited Jun 1 '20 at 6:20](#)

answered Jun 1 '20 at 5:32



[pigeonhands](#)

1,480 • 13 • 15

1

Awesome, this makes sense conceptually and is a useful mental model. Thank you!

– [Kerastan](#)

[Jun 1 '20 at 6:17](#)

[Add a comment](#)

Your Answer

Post Your Answer

By clicking "Post Your Answer", you agree to our [terms of service](#), [privacy policy](#) and [cookie policy](#)

Not the answer you're looking for? Browse other questions tagged [rust](#) [closures](#) or ask your own question.

The Overflow Blog

- [Sequencing your DNA with a USB dongle and open source code](#)
- [Don't push that button: Exploring the software that flies SpaceX rockets and...](#)

Featured on Meta

- [Providing a JavaScript API for userscripts](#)
- [Congratulations to the 59 sites that just left Beta](#)

Related

[In PHP, what is a closure and why does it use the "use" identifier?](#)

[Why doesn't println! work in Rust unit tests?](#)

[Rust ownership rules in the context of closures](#)

[Why is the value moved into the closure here rather than borrowed?](#)

[Why does the closure take ownership of the vector here?](#)

[How do Rust closures work and how does it execute a closure?](#)

[How to give ownership inside Rust future](#)

[Do Rust closures really not need type annotations?](#)

Hot Network Questions

[What is this large long-legged orange and black insect?](#)

 What does the numbers mean in this guitar tab if they already gave the chords to play?

 Does anyone have a DG Business BASIC manual?

 Seeing oneself in an abstract painting

 Regular expressions within QGIS expressions: logical operator AND

[more hot questions](#)

 [Question feed](#)

STACK OVERFLOW

[Questions](#)
[Jobs](#)
[Developer Jobs Directory](#)
[Salary Calculator](#)
[Help](#)
[Mobile](#)

PRODUCTS

[Teams](#)
[Talent](#)
[Advertising](#)
[Enterprise](#)

COMPANY

[About](#)
[Press](#)
[Work Here](#)
[Legal](#)
[Privacy Policy](#)
[Terms of Service](#)
[Contact Us](#)
[Cookie Settings](#)
[Cookie Policy](#)

STACK EXCHANGE NETWORK

[Technology](#)
[Culture & recreation](#)
[Life & arts](#)
[Science](#)
[Professional](#)
[Business](#)
[API](#)
[Data](#)

[Blog](#)
[Facebook](#)
[Twitter](#)
[LinkedIn](#)
[Instagram](#)

site design / logo © 2021 Stack Exchange Inc; user contributions licensed under [cc by-sa](#). rev 2021.12.22.41046