



Products

## Use function pointer parameter inside closure Rust

Log in Sign up

Ask Question

Asked 5 years, 8 months ago

Active 5 years, 8 months ago

Viewed 525 times



2



1



I am looking to create a function, `agg`, which takes, as a parameter, another function, `get_id`, and returns an `FnMut` closure that uses the `get_id` function.

Concrete example:

```
struct CowRow {
    pub id: i32,
}
impl CowRow {
    fn get_id(&self) -> i32 { self.id }
}

pub fn agg<F>(col: F) -> Box<FnMut(&CowRow) -> i32>
    where F: Fn(&CowRow) -> i32 {
    let mut res = 0;
    Box::new(move |r| { res += col(&r); return res })
}

fn main() {
    let mut cow = CowRow { id: 0 };
    let a = agg(CowRow::get_id);
    a(&cow);
}
```

Which produces the error:

the parameter type 'F' may not live long enough [E0310]

run ``rustc --explain E0310`` to see a detailed explanation

consider adding an explicit lifetime bound 'F: 'static'...

...so that the type ``[closure@main.rs:23:14: 23:53 col:F, res:i32]`` will meet its required lifetime bounds

The idea here is that I want a generic function that allows for creating closures which operate on different fields in the struct. So, my thought was to pass a function in that is a getter for the struct and use this in the closure to extract the appropriate field.

I've tried various combinations of adding `'static` to the `agg` signature but I'm not sure what that actually means and where it would need to go syntactically. Additionally, I've tried a number of techniques from: <https://github.com/nrc/r4cpp/blob/master/closures.md> such as adding the `get_id` method as a trait but have been unable to get that working either.

[closures](#) [rust](#) [function-pointers](#) [lifetime](#)

Share

Improve this

question

Follow

asked Apr 9 '16 at 16:30



josh

1,406 ● 1 ● 15 ● 27

Add a comment

1 Answer

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



6



The type parameter `F` to your function has an associated lifetime (just like every other type). But implicitly, the return value of your function, `Box<FnMut(&CowRow) -> i32>`, is really `Box<FnMut(&CowRow) -> i32 + 'static>`. That is, unless you specify a lifetime for a box, it assumes its contents can live forever. Of course if `F` only lives for `'a`, then the borrow checker is going to complain. To fix this, either

- Force `F` to have a static lifetime so that it can live forever inside the box ([playpen](#)):

```
fn agg<F>(col: F) -> Box<FnMut(&CowRow) -> i32>
    where F: Fn(&CowRow) -> i32 + 'static
```

**Your privacy**

By clicking "Accept all cookies", you agree Stack Exchange can store cookies on your device and disclose information in accordance with our [Cookie Policy](#).

...

Accept all cookies } Customize settings

- Explicitly state that `F` has lifetime `'a` and so does the `Box` ([playpen](#)):

```
fn agg<a, F>(col: F) -> Box<FnMut(&CowRow) -> i32 + 'a>
where F: Fn(&CowRow) -> i32 + 'a
{
  ...
}
```

The second version is more general than the first and will accept more closures as arguments.

[Share](#)

[Improve this answer](#)

[Follow](#)

answered Apr 9 '16 at 19:08



[Dzin](#)

1,028 • 7 • 11

[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 [closures](#) [rust](#) [function-pointers](#) [lifetime](#) 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

[What's the nearest substitute for a function pointer in Java?](#)

[JavaScript closure inside loops – simple practical example](#)

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

[What's the real meaning of the error "closure may outlive the current function"?](#)

[How to add lifetime argument to closure not returning a reference](#)

[0](#)  
[Pass a Struct<a>::method as a 'for<a> Fn\(&Foo<a>\)' for use in a closure](#)

[Does move on a closure copy the reference "pointer" or the actual object referenced?](#)

[Rust says the function parameter does not live enough, even though proper lifetime have been placed](#)

[Trouble with Rust lifetimes: impl uses anonymous lifetime and function uses <a>, changing the lifetimes to match leads to an error](#)

#### Hot Network Questions

- [What's the social meaning of "He was a student of..."?](#)
  - [Can a Pyromancer Sorcerer deal fire damage to a creature under the effect of Invulnerability?](#)
  - [Have there been no deaths due to omicron in Africa?](#)
  - [How to convince clan leaders and Party Cadres to give up their power?](#)
  - [Does saying "Keep it up" put me in an authoritative position?](#)
- [more hot questions](#)

## 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](#)