Stack Overflow

1. About
2. Products
3. For Teams

1. Stack Internal Implement a knowledge platform layer to power your enterprise and AI tools.
2. Stack Data Licensing Get access to top-class technical expertise with trusted & attributed content.
3. Stack Ads Connect your brand to the world's most trusted technologist communities.
4. Releases Keep up-to-date on features we add to Stack Overflow and Stack Internal.
5. About the company Visit the blog

Loading…

current community

Stack Overflow

help chat

- Meta Stack Overflow

your communities

Sign up or log in to customize your list.

more stack exchange communities

company blog

3. Log in
4. Sign up

AI Assist is now on Stack Overflow. Start a chat to get instant answers from across the network. Sign up to save and share your chats.

Home
Questions
AI Assist
Tags
Challenges
7. Chat
Articles
Users
Companies
Collectives

Communities for your favorite technologies. Explore all Collectives

Stack Internal

Stack Overflow for Teams is now called **Stack Internal**. Bring the best of human thought and AI automation together at your work.

Try for free Learn more
Stack Internal

Bring the best of human thought and AI automation together at your work. Learn more

Collectives™ on Stack Overflow

Find centralized, trusted content and collaborate around the technologies you use most.

Learn more about Collectives

Stack Internal

Knowledge at work

Bring the best of human thought and AI automation together at your work.

Explore Stack Internal

Asked 4 months ago

Modified 4 months ago

Viewed 110 times

3

I'm trying to return an async future from a method on a struct that implements a trait. Here's a simplified version of my Rust code:

```
use std::future::Future;

fn main() {}

fn sleep() -> impl Future + 'static {
    let scheduler = Tokio;
    scheduler.sleep()
}

trait Scheduler {
    fn sleep<'a>(&'a self) -> impl Future + 'static;
}

struct Tokio;

impl Scheduler for Tokio {
    fn sleep<'a>(&'a self) -> impl Future + 'static {
        async {}
    }
}
```

This fails to compile with the following error:

```
error[E0597]: `scheduler` does not live long enough
--> src/main.rs:5:5
  |
4 |     let scheduler = Tokio;
  |         --------- binding `scheduler` declared here
5 |     scheduler.sleep()
  |     ^^^^^^^^^--------
  |     |
  |     borrowed value does not live long enough
  |     argument requires that `scheduler` is borrowed for `'static`
6 | }
  | - `scheduler` dropped here while still borrowed
```

I don't understand why this happens. The future returned from `sleep()` is 'static, and scheduler is just a local variable. But Rust says it's borrowed for `'static`.

How can I fix this? My goal is for `sleep()` to return a future that's `'static` so it can be spawned or used elsewhere.

- rust

Share

Improve this question

Follow

edited Jul 30 at 13:11

cafce25

30k55 gold badges4949 silver badges6868 bronze badges

asked Jul 29 at 8:03

Yanni Wang

89122 gold badges1212 silver badges2626 bronze badges

Add a comment |

1 Answer

5

`impl Trait` always captures all lifetime (and type) parameters in scope. Saying `+ 'static` doesn't help, unfortunately.

The real fix will be to use the `use<>` syntax, which was created precisely for this:

```
trait Scheduler {
    fn sleep<'a>(&'a self) -> impl Future + use<Self>;
}

struct Tokio;

impl Scheduler for Tokio {
    fn sleep<'a>(&'a self) -> impl Future + use<> {
        async {}
    }
}
```

Share

Improve this answer

Follow

answered Jul 29 at 9:05

Chayim Friedman

76.2k55 gold badges9797 silver badges140140 bronze badges

Sign up to request clarification or add additional context in comments.

3 Comments

Add a comment

Yanni Wang

Yanni Wang Jul 29 at 9:20

I got this error with your code. I am using Rust 2024. Does it work for you? `txt error: `use<...>` precise capturing syntax is currently not allowed in return-position `impl Trait` in traits --> src/main.rs:11:45 | 11 | fn sleep<'a>(&'a self) -> impl Future + use<Self>; | ^^^^^^^^^ | = note: currently, return-position `impl Trait` in traits and trait implementations capture all lifetimes in scope = note: see issue #130044 <https://github.com/rust-lang/rust/issues/130044> for more information`

2025-07-29T09:20:53.92Z+00:00

0

Reply

- Copy link

Chayim Friedman

Chayim Friedman Jul 29 at 9:41

Weird, it works for me: play.rust-lang.org/…. Is your Rust version up-to-date?

2025-07-29T09:41:49.467Z+00:00

0

Reply

- Copy link

Yanni Wang

Yanni Wang Jul 29 at 12:38

Just updated the Rust version. It works! Thank you!

2025-07-29T12:38:58.987Z+00:00

0

Reply

- Copy link

Your Answer

Thanks for contributing an answer to Stack Overflow!

- Please be sure to *answer the question*. Provide details and share your research!

But *avoid* …

- Asking for help, clarification, or responding to other answers.
- Making statements based on opinion; back them up with references or personal experience.

To learn more, see our [tips on writing great answers](#).

Start asking to get answers

Find the answer to your question by asking.

[Ask question](#)

Explore related questions

- [rust](#)

See similar questions with these tags.

- The Overflow Blog
  [Introducing Stack Overflow AI Assist—a tool for the modern developer](#)
  [Treating your agents like microservices](#)
- Featured on Meta
  [Chat room owners can now establish room guidelines](#)
  [AI Assist is now available on Stack Overflow](#)
  [Policy: Generative AI (e.g., ChatGPT) is banned](#)

Related

Hot Network Questions

- [Download is performed unsandboxed as root - Why am I getting this message, and how do I fix it?](#)
- [Is it possible to use LaTeX to create RPG-related PDFs, such as character sheets or adventure modules?](#)
- [Is there any way to restore hearts without consuming anything, without hot springs, and without shrines?](#)
- [Curve has weird bumps when triangulated](#)
- [Does short range teleporting end floating disk?](#)
- ["Creatures your opponents control" after the ability is activated](#)
- [TV program where real actors interact with 2-Dimensional characters](#)
- [What does Felix mean by telling Bond to pick a contact point while "standing up" in Diamonds Are Forever?](#)
- [How does supporting "One China" work without also supporting Taiwan-China reunification?](#)
- [The first Pontrjagin class of Milnor's example of exotic 7-sphere](#)
- [Trying to understand UK tax brackets for salary above 150k](#)
- [When did "doubt" meaning question or point of uncertainty fall out of use outside of Indian English?](#)
- [Vertical alignment of equations inside tabularx](#)
- [Preservation of universal sentences under substructures in higher-order logic](#)
- [When the fuel governor fails, why is autorotation the only option for landing?](#)
- [Is there any point in sharing a scientific discovery as a "nobody"?](#)
- [Who is Patrick and why is he referred to at anti-AfD demonstrations?](#)
- [Ping sweep of IP subnets](#)
- [How can I secure a loose kitchen cabinet?](#)
- [Is "We will review application starting from..." a deadline?](#)
- [Should these weep hole covers be removed from the exterior of sliding glass door?](#)
- [Would it be plausible to have a flat eye with a moving pupil?](#)
- [Why is "cmd | xargs" to join lines of cmd with spaces wrong](#)
- [Correct website for Thailand ETA application form](#)

[more hot questions](#)

[Question feed](#)

Subscribe to RSS

Question feed

To subscribe to this RSS feed, copy and paste this URL into your RSS reader.

> [   ]

Stack Overflow

- [Questions](#)
- [Help](#)

- Chat

Business

- Stack Internal
- Stack Data Licensing
- Stack Ads

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