

[Structure with a field whose lifetime depends on a second field](#)

[help](#)

[pixel](#) February 24, 2016, 1:03pm 1

This is basically the code I would like to implement:

```
use std::boxed::Box;

struct X<'a> {
    data: &'a String
}

struct Y {
    data: String
}

impl Y {
    fn create<'a>(&'a self) -> X<'a> {
        X { data: &self.data }
    }
}

struct Z<'a> {
    y: Box<Y>,
    x: X<'a>
}

impl<'a> Z<'a> {
    fn new(y: Box<Y>) -> Z<'a> {
        let x = y.create();
        Z { y: y, x: x }
    }

    fn destroy(self) -> Box<Y> {
        self.y
    }
}

fn main() {
    let y = Box::new(Y { data: "foo".to_string() });
    let z = Z::new(y);
}
```

Structures X and Y come from a 3rd party library so I cannot change them. I would like to create structure Z. Putting Y in a box is not required, I just through I might be able to get away with more if it was.

What I'd really like to do is implement Drop for Z so that Y is passed back to a collection of Ys. So I think i'm going to actually have to use Box::into_raw and Box::from_raw so that the destruction of Z doesn't destroy Y.

However I still have the problem of the lifetime on Z, since X needs a lifetime, Z needs a lifetime, but I don't know how to set Z's lifetime.

[gkoz](#) February 24, 2016, 1:15pm 2

If Z wants to borrow something from Y I don't think it makes sense to also own Y. It could borrow Y though:

```
struct Z<'a> {
    y: &'a Y,
    x: X<'a>,
}

impl<'a> Z<'a> {
    fn new(y: &Y) -> Z {
        let x = y.create();
        Z { y: y, x: x }
    }
}
```

}

[oli_0bk](#) February 24, 2016, 2:09pm 3

There's a gazillion posts on stackoverflow. The main one is [rust - Why can't I store a value and a reference to that value in the same struct? - Stack Overflow](#) I think.

[pixel](#) February 24, 2016, 3:10pm 4

I guess I'm going to have to restructure how this code is written. Really I want an instance of X in a thread local variable, since it's needed in a bunch of locations that are far apart in the call stack, but I may just have to keep passing it down the stack.

Related topics

Topic	Replies	Views	Activity
Lifetime issue for a struct X (optionally) owning another struct Y that has a reference to X	6	881	January 12, 2023
Strange lifetime issue with reference to another struct member	5	338	July 30, 2023
Need help storing a value with lifetime in a struct	6	1348	February 12, 2023
Lifetime in future confusion	4	543	October 7, 2021
Help regarding lifetime help	5	570	July 11, 2019

- [Home](#)
- [Categories](#)
- [Guidelines](#)
- [Terms of Service](#)

Powered by [Discourse](#), best viewed with JavaScript enabled