



Products

## How to implement a struct in Rust that has a list of itself as a field

[Log in](#) [sign up](#)

[Ask Question](#)

Asked 1 year, 6 months ago

Active 1 year, 6 months ago

Viewed 443 times



0



I have started out learning Rust and is currently trying to write a small neural network as personal exercise. I want to define a struct for my forthcoming Layers/Clusters/Groups of nodes. My initial definition looks like this:

```
struct Layer {  
    name: String, // Human readable name  
    id: String, // UUID in the future  
    order: u8, // int for sorting  
    width: u8, // Number of nodes  
    input: [&Self], // References to other Layers that feed input into this  
}
```

The thing I am struggling with is the `input` field which should contain a list of references to other Layer-instances. I will know at compile time how many each Layer will have in the list so it wont have to be mutable. Is it possible to do this? I cant find a solution on the Google machine or in "the book".

Please advise.

[struct](#) [rust](#)

[Share](#)

[Improve this](#)

[question](#)

[Follow](#)

asked Jun 17 '20 at 6:38



JMDE

895 • 9 • 14

[Add a comment](#)

1 Answer

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



4



Is it possible to do this? I cant find a solution on the Google machine or in "the book".

Possible yes, though I would not recommend it.

Let's start with the possible: `&Self` would be a "layer reference" with an unnamed lifetime, a lifetime name is for the form `'<symbol>`, so when you write `&Self` you're specifying a reference of lifetime `'Self`, but you're never specifying the *type* being referred to, which is why rustc complains about "expected type".

If you add a "proper" lifetime name, and parametrize the structure, it compiles fine:

```
struct Layer<'sublayers> {  
    name: String, // Human readable name  
    id: String, // UUID in the future  
    order: u8, // int for sorting  
    width: u8, // Number of nodes  
    input: [&'sublayers Self], // References to other Layers that feed input into this  
}
```

However I would not recommend it as the last member being a slice means it's a [DST](#) which are difficult to work with at the best of time -- as the nomicon specifically notes "custom DSTs are a largely half-baked feature for now".

Since Rust doesn't yet have const generics proper you can't use an array you'd parameterize through `layer` either (e.g. `Layer<const Size>` and `input: [&Self;Size]`, maybe one day), so you probably want something like a vector or a slice reference e.g.

```
struct Layer<'slice, 'sublayers: 'slice> {  
    name: String, // Human readable name  
    id: String, // UUID in the future  
    order: u8, // int for sorting  
    width: u8, // Number of nodes  
    input: [&'slice [&'sublayers Self], // References to other Layers that feed input into this  
}
```

**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](#)

[Share](#)

[Improve this answer](#)

Follow  
edited Jun 17 '20 at 7:18

answered Jun 17 '20 at 7:10



Masklinn

20.6k • 1 • 17 • 34

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 [struct](#) [rust](#) 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

[How to initialize a struct in accordance with C programming language standards](#)

[Is there a faster/shorter way to initialize variables in a Rust struct?](#)

336

[How do I print the type of a variable in Rust?](#)

[How to print struct variables in console?](#)

[How to disable unused code warnings in Rust?](#)

[How do I split a string in Rust?](#)

[Why can't I store a value and a reference to that value in the same struct?](#)

[How to implement rustc\\_serialize::Decodable for struct that has a field value that is another struct?](#)

[Why does the Rust compiler not optimize code assuming that two mutable references cannot alias?](#)

[How to have multiple references for a single node in a tree structure using Rust](#)

Hot Network Questions

- What edges are not in a Gabriel graph, yet in a Delauney graph?
- what is the official procedure if you answer "yes" to any of the US visa (DS 160) "are you a terrorist/slaver/bodysnatcher" questions?
- Problems that are polynomially "hard" to compute but "easy" to verify
- How to salvage bitter homemade mustard?
- Most notable papers in Economics in 2021

[more hot questions](#)

Question feed

[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