

Have you considered *Rewriting It In Rust?*

My first proper programming language was C. It was super confusing, and I wrote some terrible code that didn't really do anything terribly useful.

Along the way, however, I learned a bit about how computers do things, sort of. I say sort of because [C is a high level language](#).

Of course, there are *higher-level* languages, and those end up being really good for a very wide array of tasks, so I went and learned them, and then I started actually getting things done. I'm glad I started with C, though, since it gave me an idea of what these other languages must be doing under the hood to accomplish so much with so little code.

Using Scheme, Python, or Javascript, I didn't have to bother with managing the computer's memory allocation manually, and that turned out to be a lot more fun.

At some point a while back, I heard about [Rust-lang](#).

What is Rust?

Rust is a systems programming language that runs blazingly fast, prevents segfaults, and guarantees thread safety.

Sounds good, right?

Yea, actually.

I think it's a pretty awesome language. The problem is, I make my living writing stuff working on web apps, and that means writing Javascript. That's not a problem, necessarily, but I just don't have much use for a language like Rust.

It definitely seems awesome, and as a lisp aficionado, I love a lot of its high level concepts. As I recall, Rust even has properly nesting block comments, which I find quite practical.

Where am I going with this?

Well, I just wanted to clear up that I don't think Rust is a bad language. I'm not going to criticize *Rust* in this article, however, I am going to criticize a part of the community surrounding the language.

To be honest, actually, I'm not really sure the people I'm going to criticize are even a part of the community. At best, they seem to be a part of the fringe.

Who are these people?

They are the people who show up unannounced on your software project's doorstep (probably your issue tracker) and ask the following question:

| Have you considered ***Rewriting It In Rust?***

Yes, I'm starting a language flame war

Ok, here we go.

If you've spent any time at all in a community where programmers hang out, you've probably seen a flame war over which language is best.

I don't know which language is best, though I've argued for Scheme many times because nobody else seems to be doing it.

That being said, I've noticed that this particular subset of the Rust community seems to do this particular thing often enough that it's nearly the only thing

that I know about the Rust community.

I've joked about it with friends and coworkers, and sure enough, most of them have witnessed it in the wild.

I believe the first time I saw it was on the (now disabled) [cjdns](#) issue tracker.

I was lucky enough to have the tracker turned on for long enough that I was able to get a screenshot:

Eventually Moving to Rust #517

 **Open** **systemkor** opened this issue on May 3, 2014 · 5 comments



systemkor commented on May 3, 2014



This is a bit off topic but I was wondering if this project would ever be interested in moving to Rust were it has high emphasis on doing things right with memory and have concurrency built in hopefully prevent thread problems and still allows for custom garbage collection.



 **cjdelisle** commented on May 5, 2014

Owner



```
user@toshitba:~/wrk/cjdns$ find ./ -wholename "**/dependencies/**" -prune -o -name '*.[ch]' -exec cat {} \; |  
wc -l  
50051
```

so uhh if you write 50 lines of code per day
(that's actually pretty fast because this stuff is nasty math-heavy code)
and you work 7 days a week...
then it'll be in Rust in about two and three quarters years.

Have fun,
Caleb

...



systemkor commented on May 6, 2014



Very true. The reason I was asking was that what is the long term goal on robust/secure coding.

It all started and ended pretty quickly. Some *Rustafarian* asked a question, and the author ([@cjdelsile](#)) answered somewhat sarcastically.

Once OP figures out that their brilliant idea is not being taken seriously, they get annoyed and leave.

But wait, there's more

Yea, of course it goes on. I heard it a few more times, laughed about it (oh, this again), and then continued on with my life.

Of course, clearly it happened enough times that I started to notice a trend, because as I mentioned, I've joked about it with people, and they've noticed it too.

So, this morning (**2016-03-22**) I saw [a thread](#) in a repo I'm watching, and there it was again:



chpio commented 4 days ago



hi,

it's maybe an odd question but^^ why is this (will be) written in go (aaaand not eg rust)?

just a personal preferences of the dev(s)?



ehmry commented 4 days ago

fc00::8 member



Why haven't you written it in rust?



chpio commented 4 days ago



hmm, my actual question was about building a maybe more valuable solution (rust can run in more environments due to the minimalistic runtime) and combining the effort and not splitting it up by maintaining multiple implementations.

also im a rust n00b



chpio commented 4 days ago



[ipfs/go-ipfs#2467](#) ipfs is "more high level", so it's "okay" if it is not that "minimalistic". But lowlevel stuff like cjdns/fc00 should be able to start without allocating hundreds of MB (imho).



ehmry commented 4 days ago

fc00::8 member



It wouldn't make any sense to write go-fc00 in Rust, that would just be too confusing.

Maybe you want <https://github.com/Randati/cjdrs>.



Iglert commented 3 days ago

fc00::8 member



I wanna integrate it into go-ipfs so rust is not an option right now. @chpio I'd be happy if you wrote it in Rust ;)



Iglert closed this 3 days ago

Once again, my friends make some snappy comments. Pair those with a classic *if you want it written why don't you write it*, a few more *lulz* and our *Rustacean*

acquaintance gets annoyed:

ok, didn't thought a simple question would attract that much angry trolls.

..and things pretty much die out.

But then I started thinking

We joked about it on IRC, and a few more people comment that they've also seen this in the wild, and quickly enough we find that we need an acronym because *Rewrite It In Rust* takes too long to type and so **RIIR** is born.

So, naturally, once an acronym is born, we have a meme, and memes like to spread. So, naturally, I start digging:

Who else has been hit by this storm that has been sweeping the web?

Nix

Apparently there were a few threads like [this one](#) which went on for a while. I haven't checked the exact timelines, but since this is a language flame war, I'm sure somebody will leap in to do that research for me.

[Nix-dev] Perl -> C++

jeaye [contact at jeaye.com](mailto:jeaye@jeaye.com)

Wed Dec 30 18:04:10 CET 2015

- Previous message: [\[Nix-dev\] Perl -> C++](#)
- Next message: [\[Nix-dev\] Perl -> C++](#)
- **Messages sorted by:** [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#)

On Wed, Dec 30, 2015 at 04:13:08PM -0500, Kevin Cox wrote:
> On Dec 30, 2015 21:21, "Arseniy Seroka" <[\[1\]grs.seroka at gmail.com](mailto:1grs.seroka@gmail.com)> wrote:
>
> In **my opinion** go is not a good language. And lua is not a language
> that is needed in that kind of program.
>
>
> I have to agree with this. Rust is awesome for it's strong type system but
> c++ is so standard that it might be worth sticking with it.

As C++ devs, I think it's important that we consider using a safer, more functional language everywhere we can. Rust does this very well, to the point where enumerating its benefits would be redundant. The point I want to make is that, if there's ever a choice between C++ and Rust, and some niche features like template meta-programming aren't required, Rust is the safer, more modern option.

If Nix is ever to adopt Rust, as a C++ replacement or otherwise, there's no better time than the present.

-
- Previous message: [\[Nix-dev\] Perl -> C++](#)
 - Next message: [\[Nix-dev\] Perl -> C++](#)
 - **Messages sorted by:** [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#)

[More information about the nix-dev mailing list](#)

Tor

Yup, Tor, the very large, critical infrastructure that keeps journalists, criminals, schizophrenics, and piracy privacy activists anonymous has been [asked to rewrite their software](#).

[Search](#)[Login](#) | [Preferences](#) | [Help/Guide](#) | [About Trac](#) | [Register](#)[View Tickets](#)[Browse Source](#)[Roadmap](#)[Timeline](#)[Wiki](#)[Search](#)[Tags](#)[← Previous Ticket](#) | [Next Ticket →](#)#11331 **new project**

Opened 2 years ago

Last modified 9 months ago

rewrite tor in rust-lang

Reported by:	cypherpunks	Owned by:	
Priority:	Medium	Milestone:	Tor: very long term
Component:	Tor	Version:	
Severity:		Keywords:	
Cc:	gk	Actual Points:	
Parent ID:		Points:	
Reviewer:		Sponsor:	

Description

Rust provides many features that would benefit Tor like memory safety, etc. Writing C code that is safe is super hard (even for experienced Tor devs).

▼ Child Tickets

▼ Change History (6)

 Oldest first
 Newest first
 Comments only

Changed 2 years ago by nickm

comment:1

- Milestone set to [Tor: very long term](#)

Not likely to happen except as an effort to refactor Tor into different modules and rewrite piece-by-piece. OTOH, there's nothing stopping somebody from doing a compatible implementation in whatever language they like. Rust enthusiasts: Prove the superiority of your language by writing a simpler, better, bug-free Tor today! ;)

Changed 9 months ago by frewsxcv

comment:2

I'm really interested in getting the ball moving for this. I'm going to look how the Mozilla devs integrated Rust into Firefox. Also, like nickm said in the previous comment, this should be done module

More still...

Why not [rewrite the Linux Kernel](#)? I'd love to hear Linus's answer to this.

Then of course there's [this comprehensive thread](#) on Hacker News covering a bunch of angles on the pros and cons of this meme.

Finally, I stumbled across [this blog](#).

Despite being quite new, and certainly newer than the meme itself, this post seems to exemplify many of the talking points extolled by proponents of **RIIR**.

Its title even comes in the form of a call to action:

| Rewrite Everything In Rust

So...?

I don't really have more to say on the matter. I just wrote this much to document a phenomenon I've noticed.

I don't have a comments section, because it ends up being really time consuming to say annoying, opinionated things if you have to answer for the fallout in your comments section. Instead, I made [a github repo](#) where you can [file an issue](#) which:

1. voices your own obnoxious opinion
2. supports my thesis
 - super bonus points if you find more github issues asking for Rust rewrites
3. refutes my thesis
 - super science points if you find that this behaviour is not exhibited disproportionately by Rust users
4. links to some project where somebody actually did *Rewrite It In Rust*

Did this post offend you?

Cool.

Talk about it.

Tell your friends.

Good memes die hard.

--ansuz

2016-03-22