

Rust Limericks

Andre 'lloqig' Bogus

Rust-land is a very strange place
sans null deref nor data race
it has its own styles
but once it compiles
it will not blow up in your face

In Rust-land, much unlike in C
borrowck looks out for me.
It checks all my borrows
'gainst aliasing sorrows
and lets me code concurrently.

The Rust borrow
checker will prove
that no mutable
reference coexists
with another
reference, thus
ensuring memory
safety

Rustaceans are very polite
read the Code of Conduct each night
We value the lot
including each bot
and don't shout 'WHO SENT ME THIS
SHIT?'

The Rust Community
follows the Code of
Conduct, which

The Rust community
is known to be polite,
and befriend bots

Linus Torvalds, Kernel
head honcho, lashes
out against one of his
lieutenants

For fast stuff, please code in Rust
Or do it in C if you must,
But do it correct
Or your data is wrecked.
Egad! get it working or bust.

In Rust-land we're for code reuse
We take a stand against abuse
We're having good fun
While getting stuff done
Come join us now if you so choose

Sometimes your Rust code looks weird
Because many sigils appeared
Yet the reason behind
is not hard to find
So really it is to be cheered

As gladly for Rust I will reach
it's a language I don't want to teach
far too complex-no joke
and it doesn't evoke
mental images of walks on the beach

With Rust you're getting Results
unlike any lang-based cults
Where the right way to code
lets yer neurons erode
adding to injury the insults

Need to save memory and don't know how?

You want to consider a Cow

so instead of a clone

you can borrow or own

So use it all in your code now!

I'm coding in Java by day
sometimes it gets in my way
where in Rust I abstract
I need to counteract
runaway patterns joining the fray

Java is known to be
quite pattern-happy,
sometimes leading to
overengineered code

Some new rustic RFCs landed
The process is quite heavy-handed
Someone's coding the code
and have crater explode
then the RFC gets amended

A tool to check
language changes
against all known
crates on crates.io

Aaron Turon gave his rapport
In matters of urgent import.
He's asking someone
Let his project go on,
I.e. he seeks crossbeam support.

On Rust's internal
forum, a request for
help with the
crossbeam project

(with apologies to Pokémon fans)

As many a Rust newbie feared
a wild borrowck appeared.

It's super effective

A reference detective
so your code becomes well-engineered.

Let x be a mutable slice
of T where T has a size
then we can `mem::replace(..)`
it in various ways
I think that is awesomely nice.

A method to replace
the contents of an
item in memory,
returning the original
value

Let a Rust crates bloom
any day our crater sans boom
the system be healthy
and useful-code-wealthy
despite trolls spelling impending doom

Who knows what Rust in six weeks
give us new features and tweaks
will it blend? will it slice?
and be awesome and nice?
Now I sound like a total geek.

Rust releases a new
stable and beta
version every six
weeks

Burntsushi when asked to rewrite
ringrep in C++ said he might
but he'd like to refrain
from the need to maintain
it as that'd be hell of a fight

Andrew Gallant

The maintainer of the
benchmarksgame site
asks if ripgrep could
have been written in
C++

With Box you put stuff on the heap
it is surprisingly cheap
and unlike on stack
things stay if you go back
from a function (or otherwise leap)

[@Manishearth](#) petitioned to rid

our community of people who did

enjoy pineapple pie -

I don't quite know why

but I don't like exclusion one bit

As Rust has implicit Drop
Your data will silently plop
out of memory as one
unless you're holding on
to it until your program will stop

A trait to dispose of
values

Result is a return type tailored
to functions that either do fail or
return on success
to the calling process
viz. Rust is quite candid on failure

The Rust team took the initiative
by discussing with rustic natives
their wants and their needs
if you would ask me: Please
make next Rust concatenative

The Rust ergonomics
Initiative

The red monks just went ballistic
over their semestral statistic
As TypeScript and Rust
Gained lots of trust
that is anyway probabilistic

Use [error-chain](#) to handle faults
[rand](#) for randomness & [quickcheck](#) for naught
[rayon](#) - all cores go
[tokio](#) - async IO
Those crates are pragmatic defaults

Jake & Carol are asking around
someone up for a sponsoring round
to write a cargo clone
that runs inhouse, alone
so enterprise use may abound

Integer32 LLC: A Rust
consultancy company

[@SimonSapin asked](#) for detail
on what Systems Programming entail
so Rust is there
but Go is no fair
yet I think the distinction is frail

(Full disclosure: I misinterpreted Simon's
intention)

StackOverflow asked once more to learn
what tech we love and what we spurn
VB6 still a flop,
Rust came out on top
this year for yet another turn

The annual
StackOverflow
developer survey

Rustaceans will usually track
their heap usage forwards and back
so no allocation
lead to altercation
One could say that we code full-stack

```
for i in [1, 2].iter().map(
|x| x + 1).filter(|zap|
zap % 2 == 0) {
    println!("You are my hero");
} // ok, now I'm taking a nap
```

Our [clippy] will warn us on code
that may at some day corrode
through our source and about
it'll gladly look out
for patterns that do not well bode

Rust coding is one of the best
if you want to easily test
your code when given stuff
performs correct enough
cargo fuzz'll take care of the rest

A fuzz testing tool for
Rust code

When C++ devs discussed Rust
some of them concluded we must
reign in some people
who shout “wake up, sheeple”
(and I think their request is just)

Some people will evermore be
fascinated by K&R's C
though it will crash and burn
they do hope once to learn
to work with `malloc` and `free`

Brian Kernighan &
Dennis Ritchie,
creators of the C
programming
language

C functions for error-
prone manual
memory
management

If in Rust code you need to store
one item of some type or more,
you say: “What the heck!
Let’s simply use [Vec](#)!”
(the name is omitting a “tor”)

With Google now running the show
The zeitgeisty language is Go
However one just
can program in Rust
and tell Thompson & Pike it ain't so.

Two of Go's creators

Rust does have some likeable traits
works well with mutable states
we can safely abstract
what in C can be cracked
so the risk of pwnage abates

Traits in Rust allow to
describe behavior
Because Rust
while abstracting the
disallows mutation
data
only in concurrent
settings, it works very
well with it, unlike
some functional
languages

Any closure will - if it can
implement a trait we call Fn
also be FnMut
(and that's double plus good)
and perhaps FnOnce once again

Rust's Default trait gets more hype
the more you use it for a type
for you may wanna strive
to use auto-#[derive]
Result: You have less code to type

A trait to abstract the
notion of a default
value. Can be auto-
derived for
structs/enums if all
components in turn
implement it

Rust's match clause has sufficient might
to destructure values alright
Its arms may be guarded
its patterns regarded
conceptually really tight

“Rust will fail 'cause of its style,
borrowck and time to compile,
all that complexity
is not even in C!”
(now that's what trolls mostly revile)

You want Rust to parse - you may choose
nom, combine, chomp, LALRPOP, peruse,
nom, neel or another
there are many other
Now you still don't know which one to use

(This led to the follow-up below)

Martin Hellspong demands 1 review
each Rust parsing lib old&new
in limerick form
(my usual norm)
but it's too much work in my view.

```
let sum=args().into_iter().fold(
0,|old,p|p.parse::<u64>().unwrap()+old);
println!("{}", BADA-BUM",
0u64 + sum);
//this 1's limerick gold
```

If you implement From<T> for U
Rust will freely deliver Into
so use From if it fits
(if th'orphan rule permits)
and the rest the compiler will do

When in Rust code you use a `for`
it'll loop and each time ask for more
if it does get a `None`
iteration is done
otherwise there's `Some (value)` in store

In May Rust is two years old
The changes have been pretty bold
Every six weeks it grew
out the old,in the new
The stability promise'll hold

May 2017 marks the
two-year anniversary
of Rust's first stable
release

SHAR a game built in Rust

Got [greenlit on Steam](#) so we just

Send congratulations

amidst great celebrations

Soon we'll sate our rust gamedev lust

If someone alleges you must
rewrite your project in #rustlang
[@rustevangelism] will do
what it needs to undo
the PR gaffe&mop up the dust.

Our Rust has a problem to be
quite simple to interface C
because it is believed
C strings are ill-conceived
(besides malloc isn't free).

C strings have been
shown to be
underspecified

With Rust more folks get access
to systems programming with less
fearful footguns and stuff
(C++ has enough)
Here's to its ongoing success!

rustdoc won't hoedown no more
the markdown it had used before
instead pulldown-cmark
builds gay as a lark
too beautiful docs to ignore

The formerly used
markdown parser

A CommonMark
parser written in Rust

rustc does more things on demand
and caches things unless it can't
to faster compile
(or error in style)
the Rust code it just has on hand.

It is now a common refrain
to ask Rustaceans to abstain
from being a bother
and not ask another
C++ dev Rust coding to train.

VSCode now gets its search power

by ripgrep & won't need 1 hour

2 search through your code

4 that thingy you wrote

alone in your ivory tower

Rustaceans collectively swoon
at a young fictitious girl who soon
may be the next Steve
a few are peeved
her threads go pop like a balloon

[rayon]'s getting faster & better

It follows iters to the letter

Reuses a pool

for threads, that's so cool!

and it splits and it slices, et cetera.

Sōzu is a proxy, reverse

With stated goals quite diverse

It shouldn't be brash

Nor panic or crash

Rust willing, it'll proxy this verse.

Rust language can now be compiled
For the web, with rustfmt styled,
and with clippy be checked
So it's nice and correct
Let @rustevangelism go wild!

A 2017 April Fool's
joke: The Rust
Evangelism Strike
Force (R.E.S.F.)

Rust land is a commie utopia
A zero-cost cornucopia
For we may freely loan
things that others must own
It's like ownership's causing myopia

If into Rust you want to enter
join a project and find you a mentor
who'll help you for free
so you can easily
become a great Rust implementer

The orange site jointly discussed
a programming language called Rust
'Tis but too weird
some people feared
being trollishly non-C++ed.

Hacker News

programming in Rust is good fun
however your code will not run
unless you have shown
that you correctly own
your data 'cause that's how it's done

(At the request of Carol Nichols:)

The core team was reinforced twice

With Carol and Nick! Congrats! Nice!

Now here's in due time

your humble rhyme

That's my way of breaking the ice

Rust code doesn't need a GC
there's no delete needed, nor free
yet people complain
again and again
'bout lifetimes, how hard they'd be

Garbage Collector

If in python you're coding happy
Your code will work, but not snappy
But in Rust it will roar
Leave you asking for more
And that borrowck is a nice chappy

With `[error-chain]` you can lay back
assume that your code is on track
to handle all faults
with nice sane defaults

`[unwrap ()]` is a lazy-ass hack

`[unwrap ()]: # TODO: Result.unwrap()`

I don't know if this thing is fake
But if not, something I would take
To write Rust code
In retro-mode
(I'll still use cargo, not make)

A (probably fake)
screenshot of a
Turbo-Pascal like Rust
IDE

Rust Maps do have Entries in store
don't need double takes anymore
They're vacant or not
anyway pretty hot
if u know what to employ'em for

The `entry()` method
of maps returns a
type that represents
the empty or filled
entry for a given key

Rust's core infrastructure indeed
had for helping hands urgent need
So a team they set up
to take care of the op
and on IRC they will meet.

Rust-Internals:
Announcing the
inofficial Rust
Infrastructure Team

Austin Hicks [did a stellar job](#)
reducing Rust memory slop
by teaching rustc
to reorder fields free
(Unless `#[repr(C)]` tells it to stop)

Amidst systems programming terror
Rust will gladly help you on error
the messages nice
with useful advice
to follow right now or where'er

Unlike functional langs Rust made
a decision for mutable state
but twice at a time
you can't mutate a rhyme
lest your program befall a bad fate

You can now rustup RIS
thus reducing CPU stress
Those will now keep chilled
no more needing to build
the whole kitchen sink, just decompress

Rust Language Server

To Kyiv I'm taking a flight
thanks [Iaim](#) for the ride
at [RustFest](#) to meet
the Rust hacker elite
(and inbetween party all night)

As Rust gains copious traction
expect a befuddled reaction
by luddites who see
only their beloved C
and shun mem'ry safety's attraction

A friend of mine tried 2 use docker
it's error messages would mock'er
Such tasteless reports
and useless aborts
for me that's a literal blocker

#rustlang serde is stable! I feel
this upgrade has mainstream appeal
O-copy read things
but escaped JSON strings
And at zero \$ it's a steal

Once Rust is widely adopted
& C's userbase is coopted
We can lastly retire
R.E.S.F. & admire
the end of unsafety-we chopped it.

Rust Evangelism
Strike Force

Java wants to copy Rust's match
but as usual there's a small catch:
It needs you to type
the expression w/ type
and not be as powerful, natch

A blog post by Brian
Goetz outlining their
preliminary plans for
match in Java

Debian asked /r/rust redditors
become cargo deb package editors
and help package Rust
(as sure someone must)
a Firefox requirement sequitur

cargo is a marvelous tool
can be used by about any fool
so you can just depend
on all crates in the end
& your code will be built! Very cool

I'm stoked to see some friendly faces
in Kyiv of all the world's places,
with a ticket I'm blessed
to 2017's [RustFest]
Let's get there without data races

Rust is known to
prevent data races

TimNN went the [arduous track]
to up Rust's [LLVM] stack
he's the hero we need
for he did succeed
where most would defeated go back

[arduous track](#)

[LLVM](#)

Redox ion shell is pretty speedy
also far from memory-greedy.

But the biggest gain:

It's syntax is sane

without nonsense like, say, 'esac' or 'fi'

crates.io has 9K crates in store
and every day is getting more
So Indy's secret lair
Looks relatively bare
Yet no crate to melt Nazis here-Or?

Some Rust code newbies and I
Got in Kyiv together to try
to learn Rust, as shown
by Ashley, who'll own
the title 'knows Rust how&why'

Ashley Williams, who
held the Kyiv
RustBridge workshop

In Kyiv there was a RustFest
The talks were the absolute best!
As is my bag
full of Rust swag
So now I'll fly home for some rest.

RustFest was great fun overall
Many great folks I met in the hall
So here's what I do
and hope you'll, too
I'll see you [in Zurich this fall!](#)